Oracle Project Portfolio Management Cloud
Implementing Project Execution Management
This guide also applies to on-premise implementations

Release 8

April 2014
Contents

1 Overview
   Project Execution Management Offering: Overview .........................................................1-1
   Common Implementation: Overview ..................................................................................1-3
   Getting Started with an Implementation: Overview .........................................................1-4
   Manage Application Implementation ...............................................................................1-6

2 Common Applications Configuration: Overview
   Define Common Applications Configuration for Project Execution Management: Overview ...............................................................................................................................2-1

3 Common Applications Configuration: Define Synchronization of Users and Roles from LDAP
   User and Role Synchronization: Explained ........................................................................3-1

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

5 Common Applications Configuration: Define Approval Management for Project Execution Management
   Approval Management: Highlights ..................................................................................5-1

6 Common Applications Configuration: Define Help Configuration
   Define Help Configuration: Overview ..............................................................................6-1
   Set Help Options ..............................................................................................................6-1
   FAQs for Assign Help Text Administration Duty .............................................................6-4
   Manage Help Security Groups ..........................................................................................6-4

7 Common Applications Configuration: Maintain Common Reference Objects
13 Importing and Exporting Setup Data

Configuration Packages: Explained ................................................................. 13-1
Exporting and Importing Setup Data: Explained ......................................... 13-1
Moving Common Reference Objects ............................................................. 13-2
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. Note: Limited content applicable to Oracle Cloud implementations.</td>
</tr>
</tbody>
</table>

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

**Other Information Sources**

**My Oracle Support**

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

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

Oracle Enterprise Repository for Oracle Fusion Applications provides details on service-oriented architecture assets to help you manage the lifecycle of your software from planning through implementation, testing, production, and changes.

In Oracle Fusion Applications, you can use Oracle Enterprise Repository at http://fusionappsoer.oracle.com for:

- Technical information about integrating with other applications, including services, operations, composites, events, and integration tables. The classification scheme shows the scenarios in which you use the assets, and includes diagrams, schematics, and links to other technical documentation.
- Other technical information such as reusable components, policies, architecture diagrams, and topology diagrams.

Documentation Accessibility

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

Comments and Suggestions

Your comments are important to us. We encourage you to send us feedback about Oracle Fusion Applications Help and guides. Please send your suggestions to oracle_fusion_applications_help_ww_grp@oracle.com. You can use Send Feedback to Oracle from the Settings and Actions menu in Oracle Fusion Applications Help.
In the Project Management business process area, your enterprise can configure how you execute projects, including how to manage requirements, schedule and collaborate on tasks, staff resources, maximize utilization, resolve issues, and complete deliverables.

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, the functional areas and features that you can select when you configure the offering, and business objects and enterprise applications that are associated with the offering.

The first implementation step is to configure the offerings in the Setup and Maintenance work area by selecting the offerings, functional areas, and features that you want to make available to implement.

The following functional areas, features, and feature choices are specific to the Project Execution Management offering:
<table>
<thead>
<tr>
<th>Offering or Functional Area</th>
<th>Feature</th>
<th>Description</th>
<th>Choices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Execution Management</td>
<td>Implementation Project Task Lists</td>
<td>Indicate whether to display the minimum or a complete list of setup tasks in implementation projects.</td>
<td>• Basic: Include the minimum number of setup tasks that must be implemented to use Project Execution Management applications. • Complete: Include all setup tasks to perform configuration tasks that are common to Oracle Fusion applications and to extend the functionality of Project Execution Management applications. For example, the Complete feature choice includes setup tasks for social networking, users and security, and application preferences.</td>
</tr>
<tr>
<td>Project Execution Management</td>
<td>Local Installation of Help</td>
<td>Define help settings if help is locally installed.</td>
<td>• Yes • No</td>
</tr>
<tr>
<td>Project Execution Management</td>
<td>Access to Internet-Based Help Features</td>
<td>Set options for using Internet-based help features.</td>
<td>• Yes • No</td>
</tr>
<tr>
<td>Project Execution Management</td>
<td>Help Customization</td>
<td>Set help options for privileged users to edit help and add custom help.</td>
<td>• Yes • No</td>
</tr>
<tr>
<td>Project Execution Management</td>
<td>Custom Help Security</td>
<td>Enables addition of links to internet sites from help, and sharing of ratings.</td>
<td>• Yes • No</td>
</tr>
<tr>
<td>Project Execution</td>
<td></td>
<td>Configure how you execute projects, manage requirements, schedule and collaborate on tasks, staff resources, maximize utilization, resolve issues, and complete deliverables. Define and manage project users, project roles, and calendars.</td>
<td>• Yes • No</td>
</tr>
</tbody>
</table>
Project Resource Management | Indicate whether Oracle Fusion Project Resource Management will be deployed. | • Yes  
• No

Next, create one or more implementation projects for the offerings, functional areas, and features 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 the **Complete** feature choice for the Implementation Project Task Lists feature, and enable all additional features, the generated task list for this offering will contain the groups of tasks listed in the following table:

<table>
<thead>
<tr>
<th>Task List</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define Common Applications Configuration for Project Execution Management</td>
<td>Define the configuration for common setup such as users, enterprise and HR structures, security, and common reference objects for Oracle Fusion Project Execution Management. You can find other information that supports the common implementation tasks in the Oracle Fusion Applications Concepts Guide.</td>
</tr>
<tr>
<td>Define Common Project Execution Options</td>
<td>Configure Oracle Fusion Project Management products to manage project users, project roles, and calendars.</td>
</tr>
<tr>
<td>Define Project Management Configuration</td>
<td>Configure Oracle Fusion Project Management to manage projects, tasks, requirements, deliverables, and resources, and to track and resolve issues.</td>
</tr>
<tr>
<td>Define Project Resource Management Configuration</td>
<td>Configure Oracle Fusion Project Resource Management to manage the availability and staffing of resources, fulfill project resource requests, and monitor resource utilization.</td>
</tr>
<tr>
<td>Define Extensions for Project Execution Management</td>
<td>Configure specific extensions for customization of Project Execution Management.</td>
</tr>
</tbody>
</table>

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

**Common Implementation: Overview**

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

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

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

**Define Common Applications Configuration Task List**

Use the Define Common Applications Configuration task list to set up and administer an implementation of behaviors across offerings.
Most Oracle Sales Cloud offerings include the Define Common Applications Configuration task list for implementing what is common in multiple or all applications available from Oracle Sales Cloud. The task lists and tasks within Define Common Applications Configuration can be present in all offerings, some, or just a single offering.

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

**Other Common Setup and Maintenance Tasks**

Other setup and maintenance tasks exist in multiple offerings but not in the Define Common Applications Configuration task list. Use these other task lists to define an Oracle Transactional Business Intelligence configuration, and to define extensions such as custom Oracle Enterprise Scheduler jobs.

You can access common implementation tasks and task lists by starting in the Setup and Maintenance Overview page and searching for task lists by name. Setup and Maintenance is available from the Administration menu to users provisioned with appropriate roles. The Administration menu provides access to other tasks, such as for customization.

**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.

- Functional Setup Manager Developer’s Guide
- Common Implementation Guide
- Customer Data Management Implementation Guide
- Enterprise Contracts Implementation Guide
- Marketing Implementation Guide
- Sales Implementation Guide
- Fusion Accounting Hub Implementation Guide
- Financials Implementation Guide
- Compensation Management Implementation Guide
- Workforce Deployment Implementation Guide
- Workforce Development Implementation Guide
- Incentive Compensation Implementation Guide
- Procurement Implementation Guide
- P6 EPPM Administrator’s Guide for an Oracle Database
- P6 EPPM Administrator’s Guide for Microsoft SQL Server Database
Implementation Projects: Explained

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

Maintaining Setup Data

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

Exporting and Importing

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

Selecting Offerings

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

Offerings: Explained

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

Implementation Task Lists

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

**Enabling Offerings**

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

**Provisioning Offerings**

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

**Options: Explained**

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

**Feature Choices: Explained**

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

You can configure feature choices in three different ways:

**Yes or No**

If a feature can either be applicable or not be applicable to an implementation, a single checkbox is presented for selection. Check or uncheck to specify yes or no respectively.
Single Select
If a feature has multiple choices but only one can be applicable to an implementation, multiple choices are presented as radio buttons. You can turn on only one of those choices.

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

In the Define Common Applications Configuration for Project Execution Management activity, you perform common setup steps such as populating the product tables containing user and role information and defining security and common reference objects for Project Execution Management applications in Oracle Fusion Project Portfolio Management. This activity contains advanced setup tasks that are not required for a typical implementation of Project Execution Management applications.

Setup tasks in the Define Common Applications Configuration for Project Execution Management activity are grouped into the following task lists:

<table>
<thead>
<tr>
<th>Task List</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define Synchronization of Users and Roles from LDAP</td>
<td>Run a process to populate the product tables containing user and role information with the users and roles held in LDAP. This process is typically the first implementation task but can also run periodically to keep the product tables synchronized with subsequent updates to LDAP.</td>
</tr>
<tr>
<td>Define Security for Project Execution Management</td>
<td>Enable users to perform functions related to their job roles.</td>
</tr>
<tr>
<td>Define Approval Management for Project Execution Management</td>
<td>Define approval routing structures and controls to match the needs of the organization.</td>
</tr>
<tr>
<td>Define Help Configuration</td>
<td>Define what users can see and do in a local deployment of Oracle Fusion Applications Help.</td>
</tr>
<tr>
<td>Maintain Common Reference Objects</td>
<td>Review and manage objects, for example currencies and reference data sets, that are shared across applications. Perform setup that applies to Oracle Fusion Applications as a whole, for example Navigator menu customizations and maintenance of common messages that can be used in any application.</td>
</tr>
</tbody>
</table>
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.

---

**Tip**

The user name and password of the installation super user are created during installation provisioning of Oracle Fusion Applications. For details of the user name and password, contact your system administrator or the person who installed Oracle Fusion Applications.
Common Applications Configuration: Define Security for Project Execution Management

Security Tasks: Highlights

Security tasks include the following.

- Security setup
- Security implementation and administration

**Note**
Security setup and administration tasks typically use integrated user interface pages that are provided by the following products.

- Oracle Identity Manager (OIM)
- Oracle Authorization Policy Manager (APM)
- Oracle Fusion Human Capital Management (HCM) products
- Oracle Application Access Control Governor (AACG) in Oracle Enterprise Governance, Risk and Compliance (GRC)

Security setup and administrative tasks performed by product administrators and implementation consultants, such as managing HCM security profiles, are presented in the documentation for those products.

**Set Up the IT Security Manager Job Role**
Provision the IT Security Manager job role with roles for user and role management.

- Using the OIM Administrator user name and password, sign in to Oracle Identity Manager (OIM). Refer to the Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management. See: Creating Users and Groups
- Open the IT Security Manager job role’s attributes and use the Hierarchy tab to add the User Identity Administrators role and the Role Administrators role in the OIM Roles category using the Add action. Use the Delegated Administration menu to search for the Xellerate Users organization and assign it to the IT Security Manager role. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.
See: User Management Tasks

Prerequisite Tasks for Security Administration

Sign into Oracle Fusion Applications for the first time with the Installation Super User account to synchronize LDAP users with HCM user management and create an IT security manager user account and provision it with the IT Security Manager role. For environments that are not in Oracle Cloud, use the super user account that was created during installation to sign in for the first time.

- Installation establishes the super user account. Refer to the Oracle Fusion Applications Installation Guide.
  See: Identity Management Configuration

- Oracle provides an initial user for accessing your services in Oracle Cloud. For more information, refer to "Oracle Cloud Application Services Security: Explained" in Oracle Cloud documentation.

- Synchronize LDAP users with HCM user management by performing the Run User and Roles Synchronization Process task. Monitor completion of the predefined Enterprise Scheduler process called Retrieve Latest LDAP Changes.

- Refer to information about creating person records in Oracle Fusion Applications Workforce Development Implementation Guide, or refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.
  See: Managing Users

- As a security guideline, provision a dedicated security professional with the IT Security Manager role as soon as possible after initial security setup and revoke that role from users provisioned with the Application Implementation Consultant role. If entitled to do so, see Security Tasks and Oracle Fusion Applications: How They Fit Together for details about provisioning the IT security manager.

Required Security Administration Tasks

Establish at least one implementation user and provision that user with sufficient access to set up the enterprise for all integrated Oracle Fusion Middleware and all application pillars or partitions.

- Perform the initial security tasks. If entitled to do so, see Initial Security Administration: Critical Choices.

- Sign in to Oracle Fusion Applications using the IT security manager’s or administrator’s user name and password, and create and provision users who manage your implementation projects and set up enterprise structures by performing the Create Implementation Users task. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.
  See: User Management Tasks

- Create a data role for implementation users who will set up HCM that grants access to data in secured objects required for performing HCM setup steps. Provision the implementation user with this View All data role. See "Creating an HCM Data Role: Worked Example."
• For an overview of security tasks from the perspective of an applications administrator, refer to the Oracle Fusion Applications Administrator’s Guide

See: Securing Oracle Fusion Applications

Optional Security Administration Tasks

Once initial security administration is complete and your enterprise is set up with structures such as business units, additional security administration tasks are optional and based on modifying and expanding the predefined security reference implementation to fit your enterprise. See points to consider for defining security, data security and trading partner security after enterprise setup.

• Create users. Refer to the Oracle Fusion Middleware User's Guide for Oracle Identity Manager.

See: Creating Users

• Provision users with roles. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

See: Adding and Removing Roles

• You manage users and job roles, including data and abstract roles, in Oracle Identity Management user interface pages. Refer to the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

See: User Interfaces

• 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

- Oracle Fusion Applications is certified to integrate with Applications Access Controls Governor (AACG) in the Oracle Enterprise Governance, Risk and Compliance (GRC) suite to ensure effective segregation of duties (SOD).
  See: Oracle Application Access Controls Governor Users Guide
  See: Oracle Application Access Controls Governor Implementation Guide

- Configure and manage auditing. Refer to the Oracle Fusion Middleware Application Security Guide.
  See: Configuring and Managing Auditing

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

After the implementation user has set up the enterprise, further security administration depends on the requirements of your enterprise. The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.

- Import Worker Users
- Import Partner Users
- Manage Job Roles
- Manage Duties
- Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human
Capital Management (HCM), Supplier Portal, and Oracle Sales Cloud Self Service, respectively, and associate them with users.

**Before Importing Users**

Oracle Identity Management (OIM) handles importing users. If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Initiate HCM Spreadsheet Load task. If user and role provisioning rules have been defined, the Initiate HCM Spreadsheet Load process automatically creates user and role provisioning requests as the workers are created.

Once the enterprise is set up, performing the Initiate HCM Spreadsheet Load task populates the enterprise with HR workers in records linked by global user ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Initiate HCM Spreadsheet Load task results in new user accounts being created. Worker email addresses as an alternate input for the Initiate HCM Spreadsheet Load task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Initiate HCM Spreadsheet Load task in HCM.

**Note**

The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in Oracle Sales Cloud.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

**Import Users**

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks. If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Initiate HCM Spreadsheet Load task to create users from imported HR workers.

Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating
data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager.

In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

**Manage Job Roles**

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

*Note*
Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, Oracle Sales Cloud, or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

**Manage Duties**

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering. OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.
In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

**Tip**
As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

**Note**
Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

**Manage Application Access Controls**

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

**Note**
In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

**Security Tasks and Oracle Fusion Applications: How They Fit Together**

The major security tasks and their order within the context of an overall Oracle Fusion Applications implementation extend from security setup through production deployment audits. The Oracle Fusion business process model (BPM) provides a sequence of security implementation tasks that includes the following.

- Security setup (Define Common Applications Configuration activity)
  - Define Implementation Users task group (optional)
    - Create Implementation Users task
    - Create Data Role for Implementation Users task
    - Provision Roles to Implementation Users task
• Define security - tasks vary depending on deployed Oracle Fusion product family
  • Revoke Data Role from Implementation Users task
  • Import Worker Users task
  • Import Partner Users task
  • Manage Duties task
  • Manage Job Roles task
  • Manage Application Access Controls task
• Define Automated Governance, Risk, and Performance Controls activity
  • Manage Application Access Controls task (AACG settings)
  • Manage Application Preventive Controls task
  • Manage Application Transaction Controls task
  • Manage Application Configuration Controls task
• User and role provisioning tasks
  • Implement Role Request and Provisioning Controls activity
  • Import Worker Users task
  • Import Partner Users task
  • Self Request User Roles task
  • Approve User and Role Provisioning Requests task
  • Assign User Roles task
  • Manage Supplier User Roles and User Role Usages task
  • Map and Synchronize User Account Details task
  • Tasks for viewing account details for self or others
  • Tasks for applying and managing various role provisioning rules
  • Tasks for running synchronization processes
• Security implementation and ongoing maintenance after setup (Manage IT Security activity)

• Implement Function Security Controls
  • Create Job Role task
  • Import Worker Users task
• Import Partner Users task
• Manage Duties task
• Manage Job Roles task
• Manage Users task
• Implement Data Security Controls
  • Manage Data Security Policies task
  • Manage Role Templates task
  • Manage Segment Security task
  • Manage Data Access Sets task
  • Define Security Profiles task group
• Auditing tasks
  • Manage Security Audit, Compliance and Reporting activity
  • Manage Application Access Controls task

**Note**
Go live deployment does not require lockdown or specific security tasks because security is enforced across the test to production information life cycle.

**Required Roles**
The following enterprise roles are provisioned to a single super user that is set up by the Oracle Fusion Applications installation process, and to the initial user set up by Oracle for Oracle Cloud Application Services:
  • Application Implementation Consultant
  • IT Security Manager
  • Application Administrators for the provisioned products
Initial security administration also includes provisioning the IT Security Manager role with Oracle Identity Management (OIM) roles for user and role management.
  • Identity User Administrator
  • Role Administrator
Additionally, the Xellerate Users organization must be assigned to the IT Security Manager role.

**Important**
As a security guideline, provision a dedicated security professional with the IT Security Manager role at the beginning of an implementation, and revoke that role from users provisioned with the Application Implementation Consultant role.

**Tools Used to Perform Security Tasks**
Security tasks are supported by tools within both Oracle Fusion Applications and Oracle Fusion Middleware.
The figure lists the tasks associated with each of the integrated products and pillars of an Oracle Fusion Applications deployment.
Security Tasks: Overview

Security tasks span multiple business processes and are performed by various roles using numerous integrated tools. The following table shows the business process model (BPM) tasks and tools used to support securing Oracle Fusion Applications.

<table>
<thead>
<tr>
<th>Example Task</th>
<th>Oracle BPM Task</th>
<th>Supporting Tools</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>View duty roles inherited by a job role</td>
<td>Manage Duties</td>
<td>Authorization Policy Manager (APM)</td>
<td>Each logical partition or pillar contains a collection of application roles, and function and data security policies.</td>
</tr>
<tr>
<td>View entitlement or policies carried by a job role</td>
<td>Manage Duties</td>
<td>APM</td>
<td>In LDAP, the policy store stores application roles and the identity store stores enterprise roles.</td>
</tr>
<tr>
<td>Add a job role to a role hierarchy</td>
<td>Manage Job Roles</td>
<td>Oracle Identity Management (OIM)</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Add a duty role to a role hierarchy</td>
<td>Manage Duties</td>
<td>APM</td>
<td>LDAP stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions.</td>
</tr>
<tr>
<td>Task</td>
<td>Component</td>
<td>Tool(s)</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create a hierarchy of enterprise (abstract, job, data) roles</td>
<td>Manage Job Roles</td>
<td>• OIM</td>
<td>The identity store in LDAP stores enterprise roles.</td>
</tr>
<tr>
<td>Create a hierarchy of (application) duty roles</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>The policy store stores duty roles. The identity store stores enterprise roles. Some duty roles may enable actions and their associated users interface features that your enterprise does not want users to perform in Oracle Fusion applications.</td>
</tr>
<tr>
<td>Create a new job role</td>
<td>Manage Job Roles</td>
<td>• OIM</td>
<td>All functions and actions in Oracle Fusion Applications that need to be secured are covered by the reference implementation. In some cases, especially with function customizations, a new duty role may be needed.</td>
</tr>
<tr>
<td>Change duty roles inherited by a job or abstract role</td>
<td>Manage Duties</td>
<td>• APM</td>
<td>The Security Reference Manuals (SRM) document the segregation of duties (SOD) policies respected within each job role.</td>
</tr>
<tr>
<td>View Segregation of Duties (SOD) policies respected by a duty role</td>
<td>Manage Application Access Controls</td>
<td>• Application Access Controls Governor (AACG) in Oracle Enterprise Governance, Risk and Compliance (GRC)</td>
<td>The 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>
<tr>
<td>View the data security policies carried by a job, abstract, and data roles</td>
<td>Manage Data Security Policies</td>
<td>• APM</td>
<td>Oracle Fusion Data Security stores data security policies in the policy store. Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</td>
</tr>
<tr>
<td>Task Description</td>
<td>Task Name</td>
<td>Products</td>
<td>Notes</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
<td>---------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create and update HCM security profiles</td>
<td>Manage Data Role and Security Profiles</td>
<td>• Oracle Fusion HCM</td>
<td>This task does not include assigning data roles to the users, which is supported by user provisioning tasks.</td>
</tr>
<tr>
<td>Create (generate) a data role</td>
<td>1. Manage Role Templates</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.</td>
</tr>
<tr>
<td></td>
<td>2. Manage Data Roles and Security Profiles</td>
<td>• Oracle Fusion HCM</td>
<td>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 provide to supplier users</td>
<td>Manage Supplier User Role Usages</td>
<td>• Supplier Portal • Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family.</td>
</tr>
<tr>
<td>Set default supplier job roles based on the set of supplier roles that are defined by performing the Manage Supplier User Roles task</td>
<td>Manage Supplier User Role Usages</td>
<td>• Supplier Portal • Sourcing</td>
<td>These tools are in the Oracle Fusion Procurement product family.</td>
</tr>
<tr>
<td>Create a new implementation user</td>
<td>Create Implementation Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>----------------------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>Import legacy users</td>
<td>• Import Worker Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Import Partner Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a new user</td>
<td>Manage Users</td>
<td>• HCM</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provision roles to a user</td>
<td>1. Provision Roles to Implementation Users</td>
<td>• OIM</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Manage Users</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Oracle Fusion HCM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Oracle Sales Cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Oracle Fusion Suppliers</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.

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.
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 Users for Project Execution Management

Securing Identities and Users: Points To Consider

Identity covers all aspects of an entity’s existence within the contexts in which it is used. The identity of an enterprise user consists of HR attributes, roles, resources, and relationships.

HR attributes include identifying information about a user that is relatively static and well understood, such as first and last name, title, and job function.

Roles are part of a user’s identity and define the user’s purpose and responsibilities.
Within identity management, resources define what a user can and does do. In an enterprise, this typically translates into what resources a user has access to, what privileges they have on that resource, and what they have been doing on that resource. Resources can be application accounts or physical devices such as laptops or access cards. The enterprise owns the resources, secures them, and manages access to the resources by managing the user’s identity and access.

Relationships establish the portion of user identities that involve organizational transactions such as approvals.

An Oracle Fusion Applications user and corresponding identity are usually created in a single transaction, such as when a worker is created in Human Resources (HR). That transaction automatically triggers provisioning requests for the user based on role provisioning rules.

User accounts for some identities that are not employees, such as partner contacts, may be created in a later transaction using an identity that is already created in the identity store. Supplier contacts are created in the Supplier Model, not HR.

Stores

Various locations store identity and user data.

Identity data consists of the following.

- HR person records
- Oracle Fusion Trading Community Model party records

In Oracle Fusion Applications, identities and users correspond one to one, but not all identities correspond to a user, and not all users are provisioned with an identity. Some identities stored in HR and Trading Community Model may not be provisioned to user accounts and therefore are not synchronized with Oracle Identity Management (OIM). For example, a contact for a prospective customer is an identity in Trading Community Model but may not be provisioned with a user account in OIM. Some users stored in the Lightweight Directory Access Protocol (LDAP) store may not be provisioned with identities. For example, system user accounts used to run Web services to integrate third party services with Oracle Fusion Applications are not associated with a person record in HR or Trading Community Model. Some identifying credentials such as name, department, e-mail address, manager, and location are stored with user data in the LDAP store.

Importing Users

You can import users or user attributes in bulk from existing legacy identity and user stores.

Your tasks may include the following.

- Create users in bulk
- Update specific attributes for all users, such as postal code
- Link users to HR or Trading Community Model persons
- Monitor progress of the import process
• Correct errors & re-import
• Export users in bulk
• Import and export users using a standard plain text data interchange format like Lightweight Data Interchange Format (LDIF)

You can reserve a specific user name not currently in use for use in the future, or release a reserved username from the reservation list and make it available for use. Between a user registration request and approved registration, Oracle Fusion Applications holds the requested user name on the reservation list, and releases the name if an error occurs in the self-registration process or the request is rejected. Self-registration processes check the reservation list for user name availability and suggest alternative names.

**Provisioning Events**

New identities, such as new hires, trigger user and role provisioning events. In addition to user creation tasks, other tasks, such as Promote Worker or Transfer Worker, result in role provisioning and recalculation based on role provisioning rules.

When an identity’s attributes change, you may need to provision the user with different roles. Role assignments may be based on job codes, and a promotion triggers role provisioning changes. Even if the change in the identities attributes requires no role assignment change, such as with a name change, OIM synchronizes the corresponding user information in the LDAP store.

Deactivating or terminating an identity triggers revocation of some roles to end all assignments, but may provision new roles needed for activities, such as a pay stub review. If the corresponding user for the identity was provisioned with a buyer role, terminating the identity causes the user’s buyer record in Procurement to be disabled, just as the record was created when the user was first provisioned with the buyer role.

**Notifications and Audits**

Oracle Fusion Applications provides mechanisms for notifying and auditing requests or changes affecting identities and users.

Oracle Fusion Applications notifies requestors, approvers, and beneficiaries when a user account or role is provisioned. For example, when an anonymous user registers as a business-to-customer (B2C) user, the B2C user must be notified of the registration activation steps, user account, password and so on once the approver (if applicable) has approved the request and the user is registered in the system.

User ID and GUID attributes are available in Oracle Fusion Applications session information for retrieving authenticated user and identity data. End user auditing data is stored in database WHO columns and used for the following activities.

• Setting up sign-in audit
• Using the application monitor
• Notifying of unsuccessful sign ins
• Sign-in audit reports

You can conduct real time audits that instantiate a runtime session and impersonate the target user (with the proxy feature) to test what a user has access to under various conditions such as inside or outside firewall and authentication level.

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

**Delegated Administration**

You can designate local administrators as delegated administrators to manage a subset of users and roles.

Delegated administrators can be internal or external persons who are provisioned with a role that authorizes them to handle provisioning events for a subset of users and roles.

For example, internal delegated administrators could be designated to manage users and roles at the division or department level. External delegated administrators could be designated to manage users and roles in an external organization such as a primary supplier contact managing secondary users within that supplier organization.

You can also define delegated administration policies based on roles. You authorize users provisioned with specific roles named in the policy to request a subset of roles for themselves if needed, such as authorizing a subset of roles for a subset of people. For example, the policy permits a manager of an Accounts Payables department to approve a check run administrator role for one of their subordinates, but prohibits the delegated administrator from provisioning a budget approver role to the subordinate.

**Credentials**

You activate or change credentials on users by managing them in Oracle Identity Management (OIM).

Applications themselves must be credentialed to access one another.

Oracle Fusion Applications distinguishes between user identities and application identities (APPID). Predefined application identities serve to authorize jobs and transactions that require higher privileges than users.

For example, a payroll manager may submit a payroll run. The payroll application may need access to the employee's taxpayer ID to print the payslip. However, the payroll manager is not authorized to view taxpayer IDs in the user interface as they are considered personally identifiable information (PII).

Calling applications use application identities (APPID) to enable the flow of transaction control as it moves across trust boundaries. For example, a user in the Distributed Order Orchestration product may release an order for shipping. The code that runs the Pick Notes is in a different policy store than the code that releases the product for shipment. When the pick note printing program
is invoked it is the Oracle Fusion Distributed Order Orchestration Application Development Framework (ADF) that is invoking the program and not the end user.

### Manage HCM Role Provisioning Rules

#### Role Provisioning and Deprovisioning: Explained

A user’s access to data and functions depends on the user’s roles: users have one or more roles that enable them to perform the tasks required by their jobs or positions. Roles must be provisioned to users; otherwise, users have no access to data or functions.

#### Role Provisioning Methods

Roles can be provisioned to users:

- Automatically
- Manually, using delegated administration:
  - Users such as line managers and human resource specialists can provision roles manually to other users.
  - Users can request roles for themselves.

For both automatic and manual role provisioning, you create a role mapping to identify when a user becomes eligible for a role.

Oracle Identity Management (OIM) can be configured to notify users when their roles change; notifications are not issued by default.

#### Role Types

Data roles, abstract roles, and job roles can be provisioned to users. Roles available for provisioning include predefined roles, HCM data roles, and roles created using OIM.

#### Automatic Role Provisioning

A role is provisioned to a user automatically when at least one of the user’s assignments satisfies the conditions specified in the relevant role-mapping definition. The provisioning occurs when the assignment is either created or updated. For example, when a person is promoted to a management position, the line manager role is provisioned automatically to the person if an appropriate role mapping exists. Any change to a person’s assignment causes the person’s automatically provisioned roles to be reviewed and updated as necessary.

#### Role Deprovisioning

Automatically provisioned roles are deprovisioned automatically as soon as a user no longer satisfies the role-mapping conditions. For example, a line
manager role that is provisioned to a user automatically is deprovisioned automatically when the user ceases to be a line manager.

Automatically provisioned roles can be deprovisioned manually at any time.

Manually provisioned roles are deprovisioned automatically only when all of the user’s work relationships are terminated; in all other circumstances, users retain manually provisioned roles until they are deprovisioned manually.

Changes to Assignment Managers

When a person’s line manager is changed, the roles of both new and previous line managers are updated as necessary. For example, if the person’s new line manager now satisfies the conditions in the role mapping for the line manager role, and the role is one that is eligible for autoprovisioning, then that role is provisioned automatically to the new line manager. Similarly, if the previous line manager no longer satisfies the conditions for the line manager role, then that role is deprovisioned automatically.

Roles at Termination

When a work relationship is terminated, all automatically provisioned roles for which the user does not qualify in other work relationships are deprovisioned automatically. Manually provisioned roles are deprovisioned automatically only if the user has no other work relationships; otherwise, the user retains all manually provisioned roles until they are deprovisioned manually.

Automatic deprovisioning can occur either as soon as the termination is submitted or approved or on the day after the termination date. The user who is terminating the work relationship selects the appropriate deprovisioning date.

Role mappings can provision roles to users automatically at termination. For example, the locally defined roles Retiree and Beneficiary could be provisioned to users at termination based on assignment status and person type values.

If a termination is later reversed, roles that were deprovisioned automatically at termination are reinstated and post-termination roles are deprovisioned automatically.

Date-Effective Changes to Assignments

Automatic role provisioning and deprovisioning are based on current data. For a future-dated transaction, such as a future promotion, role changes are identified and role provisioning occurs on the day the changes take effect, not when the change is entered. The process Send Pending LDAP Requests identifies future-dated transactions and manages role provisioning and deprovisioning at the appropriate time. Note that such role-provisioning changes are effective as of the system date; therefore, a delay of up to 24 hours may occur before users in other time zones acquire the access for which they now qualify.

Role Mappings: Explained

User access to data and functions is determined by abstract, job, and data roles, which are provisioned to users either automatically or manually. To enable a role to be provisioned to users, you define a relationship, known as a mapping, between the role and a set of conditions, typically assignment attributes such as department, job, and system person type. In a role mapping, you can select
any role stored in the Lightweight Directory Access Protocol (LDAP) directory, including Oracle Fusion Applications predefined roles, roles created in Oracle Identity Management (OIM), and HCM data roles.

The role mapping can support:

- Automatic provisioning of roles to users
- Manual provisioning of roles to users
- Role requests from users
- Immediate provisioning of roles

**Automatic Provisioning of Roles to Users**

A role is provisioned to a user automatically if:

- At least one of the user's assignments satisfies all conditions associated with the role in the role mapping.
- You select the **Autoprovision** option for the role in the role mapping.

For example, for the HCM data role Sales Manager Finance Department, you could select the **Autoprovision** option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>Finance Department</td>
</tr>
<tr>
<td>Job</td>
<td>Sales Manager</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

The HCM data role Sales Manager Finance Department is provisioned automatically to users with at least one assignment that satisfies all of these conditions.

Automatic role provisioning occurs as soon as the user is confirmed to satisfy the role-mapping conditions, which can be when the user's assignment is either created or updated. The provisioning process also removes automatically provisioned roles from users who no longer satisfy the role-mapping conditions.

**Note**

The automatic provisioning of roles to users is effectively a request to OIM to provision the role. OIM may reject the request if it violates segregation-of-duties rules or fails a custom OIM approval process.

**Manual Provisioning of Roles to Users**

Users such as human resource (HR) specialists and line managers can provision roles manually to other users; you create a role mapping to identify roles that can be provisioned in this way.

Users can provision a role to other users if:

- At least one of the assignments of the user who is provisioning the role (for example, the line manager) satisfies all conditions associated with the role mapping.
- You select the **Requestable** option for the role in the role mapping.
For example, for the HCM data role Quality Assurance Team Leader, you could select the Requestable option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies both of these conditions can provision the role Quality Assurance Team Leader manually to other users, who are typically direct and indirect reports.

If the user's assignment subsequently changes, there is no automatic effect on roles provisioned by this user to others; they retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

**Role Requests from Users**

Users can request roles when reviewing their own account information; you create a role mapping to identify roles that users can request for themselves.

Users can request a role if:

- At least one of their own assignments satisfies all conditions associated with the role mapping.
- You select the Self-requestable option for the role in the role mapping.

For example, for the Expenses Reporting role you could select the Self-requestable option and specify the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>ABC Department</td>
</tr>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

Any user with at least one assignment that satisfies all of these conditions can request the role. The user acquires the role either immediately or, if approval is required, once the request is approved. Self-requested roles are classified as manually provisioned.

If the user's assignment subsequently changes, there is no automatic effect on self-requested roles. Users retain manually provisioned roles until either all of their work relationships are terminated or the roles are manually deprovisioned.

**Immediate Provisioning of Roles**

When you create a role mapping, you can apply autoprovisioning from the role mapping itself.

In this case, all assignments and role mappings in the enterprise are reviewed. Roles are:

- Provisioned immediately to all users who do not currently have roles for which they are eligible
• Deprovisioned immediately from users who are no longer eligible for roles that they currently have

Immediate autoprovisioning from the role mapping enables bulk automatic provisioning of roles to a group of users who are identified by the role-mapping conditions. For example, if you create a new department after a merger, you can provision relevant roles to all users in the new department by applying autoprovisioning immediately.

To provision roles immediately to a single user, the user’s line manager or an HR specialist can autoprov Provision roles from that user’s account.

**Role-Mapping Names**

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

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoprovisioned Roles Sales Department</td>
<td>Mapping includes all roles provisioned automatically to anyone in the sales department</td>
</tr>
<tr>
<td>Benefits Specialist Autoprovisioned</td>
<td>Mapping defines the conditions for autoprovisioning the Benefits Specialist role</td>
</tr>
<tr>
<td>Line Manager Requestable Roles</td>
<td>Mapping includes all roles that a line manager can provision manually to direct and indirect reports</td>
</tr>
</tbody>
</table>

**Role Mappings: Examples**

Roles must be provisioned to users explicitly, either automatically or manually; no role is provisioned to a user by default. This topic provides some examples of typical role mappings to support automatic and manual role provisioning.

**Creating a Role Mapping for Employees**

You want all employees in your enterprise to have the Employee role automatically when they are hired. In addition, employees must be able to request the Expenses Reporting role when they need to claim expenses. Few employees will need this role, so you decide not to provision it automatically to all employees.

You create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

• Employee role, and select the **Autoprovision** option
• Expenses Reporting role, and select the **Self-requestable** option
You could create a similar role mapping for contingent workers called All Contingent Workers, where you would set the system person type to contingent worker.

**Note**

If the Employee and Contingent Worker roles are provisioned automatically, pending workers acquire them when their periods of employment or placements start. If they need roles before then, you create a separate role mapping for the pending worker system person type.

### Creating a Role Mapping for Line Managers

Any type of worker can be a line manager in the sales business unit. You create a role mapping called Line Manager Sales BU and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Unit</td>
<td>Sales</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
<tr>
<td>Manager with Reports</td>
<td>Yes</td>
</tr>
</tbody>
</table>

You include the Line Manager role and select the **Autoprovision** option. This role mapping ensures that the Line Manager role is provisioned automatically to any worker with at least one assignment that matches the role-mapping conditions.

In the same role mapping, you could include roles that line managers in this business unit can provision manually to other users by selecting the roles and marking them as requestable. Similarly, if line managers can request roles for themselves, you could include those in the same role mapping and mark them as self-requestable.

### Creating a Role Mapping for Retirees

Retirees in your enterprise need a limited amount of system access to manage their retirement accounts. You create a role mapping called All Retirees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Retiree</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Inactive</td>
</tr>
</tbody>
</table>

You include the locally defined role Retiree in the role mapping and select the **Autoprovision** option. When at least one of a worker’s assignments satisfies the role-mapping conditions, the Retiree role is provisioned to that worker automatically.

### Creating a Role Mapping for Sales Managers

Grade 6 sales managers in the sales department need the Sales Manager role. In addition, sales managers need to be able to provision the Sales Associate role to other workers. You create a role mapping called Sales Managers Sales Department and enter the following conditions.
In the role mapping, you include the:
  • Sales Manager role, and select the **Autoprovision** option
  • Sales Associate role, and select the **Requestable** option

**Import Worker Users**

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

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

The Define Security activity within the Information Technology (IT) Management business process includes the following tasks.
  • Import Worker Users
  • Import Partner Users
  • Manage Job Roles
  • Manage Duties
  • Manage Application Access Controls

If no legacy users, user accounts, roles, and role memberships are available in the Lightweight Directory Access Protocol (LDAP) store, and no legacy workers are available in Human Resources (HR), the implementation user sets up new users and user accounts and provisions them with roles available in the Oracle Fusion Applications reference implementation.

If no legacy identities (workers, suppliers, customers) exist to represent people in your enterprise, implementation users can create new identities in Human Capital Management (HCM), Supplier Portal, and Oracle Sales Cloud Self Service, respectively, and associate them with users.

**Before Importing Users**

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the Initiate HCM Spreadsheet Load task. If user and role provisioning rules have been defined, the Initiate HCM Spreadsheet Load process automatically creates user and role provisioning requests as the workers are created.

Once the enterprise is set up, performing the Initiate HCM Spreadsheet Load task populates the enterprise with HR workers in records linked by global user
ID (GUID) to corresponding user accounts in the LDAP store. If no user accounts exist in the LDAP store, the Initiate HCM Spreadsheet Load task results in new user accounts being created. Worker email addresses as an alternate input for the Initiate HCM Spreadsheet Load task triggers a search of the LDAP for user GUIDs, which may perform more slowly than entering user names.

In the security reference implementation, the HCM Application Administrator job role hierarchy includes the HCM Batch Data Loading Duty role, which is entitled to import worker identities. This entitlement provides the access necessary to perform the Initiate HCM Spreadsheet Load task in HCM.

Note
The Import Person and Organization task in the Define Trading Community Import activity imports the following resources, creates users, and links the resources to users for use in Oracle Sales Cloud.

- Internal employees
- Contingent workers
- External partner contacts
- Partner companies
- Legal entities
- Customers
- Consumers

If role provisioning rules have been defined, the Import Person and Organization task automatically provisions role requests as the users are created.

Import Users

If legacy users (identities) and user accounts exist outside the LDAP store that is being used by the Oracle Fusion Applications installation, the IT security manager has the option to import these definitions to the LDAP store by performing the Import Worker Users and Import Partner Users tasks.

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the Initiate HCM Spreadsheet Load task to create users from imported HR workers.

Once users exist, their access to Oracle Fusion Applications is dependent on the roles provisioned to them in OIM or Human Capital Management. Use the Manage HCM Role Provisioning Rules task to define rules that determine what roles are provisioned to users.

Importing user identities from other applications, including other Oracle Applications product lines, is either a data migration or manual task. Migrating data from other Oracle Applications includes user data. For more information about importing users, see the Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager.

In the security reference implementation, the IT Security Manager job role hierarchy includes the HCM Batch Data Loading Duty and the Partner Account
Administration Duty. These duty roles provide entitlement to import or create users. The entitlement Load Batch Data provides the access necessary to perform the Import Worker Users task in OIM. The entitlement Import Partner entitlement provides the access necessary to perform the Import Partner Users task in OIM.

**Manage Job Roles**

Job and abstract roles are managed in OIM. This task includes creating and modifying job and abstract roles, but not managing role hierarchies of duties for the jobs.

**Note**

Manage Job Roles does not include provisioning job roles to users. Provisioning users is done in OIM, HCM, Oracle Sales Cloud, or Oracle Fusion Supplier Portal.

Roles control access to application functions and data. Various types of roles identify the functions performed by users.

The Oracle Fusion Applications security reference implementation provides predefined job and abstract roles. In some cases, the jobs defined in your enterprise may differ from the predefined job roles in the security reference implementation. The predefined roles and role hierarchies in Oracle Fusion may require changes or your enterprise may require you to create new roles. For example, you need a job role for a petty cash administrator, in addition to an accounts payable manager. The security reference implementation includes a predefined Accounts Payable Manager, and you can create a petty cash administrator role to extend the reference implementation.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Enterprise Role Management Duty role, which is entitled to manage job and abstract roles (the entitlement is Manage Enterprise Role). This entitlement provides the access necessary to perform the Manage Job Roles task in OIM.

**Manage Duties**

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

Managing duties includes assigning duties to job and abstract roles in a role hierarchy using Authorization Policy Manager (APM). If your enterprise needs users to perform some actions in applications coexistent with Oracle Fusion applications, you may wish to remove the duty roles that enable those actions. For details about which duty roles are specific to the products in an offering, see the Oracle Fusion Applications Security Reference Manual for each offering.

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.
In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

**Tip**

As a security guideline, use only the predefined duty roles, unless you have added new applications functions. The predefined duty roles fully represent the functions and data that must be accessed by application users and contain all appropriate entitlement. The predefined duty roles are inherently without segregation of duty violations of the constraints used by the Application Access Controls Governor.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Role Management Duty role, which is entitled to manage duty roles (the entitlement is Manage Application Role). This entitlement provides the access necessary to perform the Manage Duties task in APM.

**Note**

Product family administrators are not entitled to create role hierarchies or manage duty roles and must work with the IT security manager to make changes such as localizing a duty role to change a role hierarchy. Setup for localizations is documented in HCM documentation.

**Manage Application Access Controls**

Prevent or limit the business activities that a single person may initiate or validate by managing segregation of duties policies in the Application Access Controls Governor (AACG).

**Note**

In AACG, segregation of duties policies are called access controls or segregation of duties controls.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Segregation of Duties Policy Management Duty role, which is entitled to manage segregation of duties policies (the entitlement is Manage Segregation of Duties Policy). This entitlement provides the access necessary to perform the Manage Application Access Controls task in AACG.

**Importing Worker Users: Explained**

You can import workers from legacy applications to Oracle Fusion Applications using the Import Worker Users task. By enabling you to bulk-load existing data, this task is an efficient way of creating and enabling users of Oracle Fusion Applications.

**The Import Worker Users Process**

Importing worker users is a two-stage process:
1. When you perform the Import Worker Users task, the Initiate Spreadsheet Load page opens. On the Initiate Spreadsheet Load page, you generate and complete the Create Worker spreadsheet. You must map your data to the spreadsheet columns and provide all required attributes. Once the spreadsheet is complete, you click Upload in the spreadsheet to import the data to the Load Batch Data stage tables.

2. As valid data rows are imported to the Load Batch Data stage tables, the Load Batch Data process runs automatically. Load Batch Data is a generic utility for loading data to Oracle Fusion Human Capital Management from external sources. This process loads data from the Load Batch Data stage tables to the Oracle Fusion application tables.

**User-Account Creation**

Oracle Fusion user accounts are created automatically for imported workers in Oracle Identity Management (OIM), unless automatic account creation is disabled.

By default, user account names and passwords are sent automatically to users when their accounts are created. This default action may have been changed at enterprise level, as follows:

- User account names and passwords may be sent to an enterprise-wide e-mail rather than to users themselves.
- Automatic sending of user account names and passwords may be disabled for the enterprise. In this case, you can notify users at an appropriate time.

**Role Provisioning**

Once user accounts exist, roles are provisioned to users automatically in accordance with current role-provisioning rules. For example, current rules could provision the employee abstract role to every worker. Role provisioning occurs automatically unless it is disabled for the enterprise.

**Importing Worker Users: Worked Example**

This example shows how to import worker users from legacy applications to Oracle Fusion Applications.

The following table summarizes key decisions for this task.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is my spreadsheet name?</td>
<td>WorkersMMDYYBatchnn.xlsx</td>
</tr>
<tr>
<td>You can define your own naming convention. In this example, the name is selected to make identifying the spreadsheet contents easy.</td>
<td>For example, Workers042713Batch01.xlsx.</td>
</tr>
<tr>
<td>What is my batch name?</td>
<td>Workers042713Batchnn</td>
</tr>
</tbody>
</table>
Summary of the Tasks

Import worker users by:

1. Selecting the Import Worker Users task
2. Creating the spreadsheet
3. Entering worker data in the spreadsheet
4. Importing worker data and correcting import errors
5. Reviewing and correcting load errors

Prerequisites

Before you can complete this task, you must have:

1. Installed the desktop client Oracle ADF Desktop Integration Add-in for Excel
2. Enabled the Trust Center setting Trust access to the VBA project object model in Microsoft Excel

Selecting the Import Worker Users Task

1. On the Overview page of the Setup and Maintenance work area, click the All Tasks tab.
2. In the Search region, complete the fields as shown in this table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Task</td>
</tr>
<tr>
<td>Name</td>
<td>Import Worker Users</td>
</tr>
</tbody>
</table>

3. Click Search.
4. In the search results, click Go to Task for the task Import Worker Users.

The Initiate Spreadsheet Load page opens.

Alternatively, you can select the Import Worker Users task from an implementation project.

Creating the Spreadsheet

1. On the Initiate Spreadsheet Load page, find the entry for Create Worker in the list of business objects.

Create Worker appears after other business objects such as departments, locations, and jobs. Those business objects must be created before worker users, regardless of how you create them.

2. Click Create Spreadsheet for the Create Worker entry.
3. When prompted, save the spreadsheet locally using the name Workers042713Batch01.xlsx.
4. When prompted, sign in to Oracle Fusion Applications using your Oracle Fusion user name and password.
Entering Worker Data in the Spreadsheet

1. In the **Batch Name** field of the spreadsheet `Workers042713Batch01.xlsx`, replace the default batch name with the batch name `Workers042713Batch01`.

2. If your data includes flexfields, click **Configure Flexfield** to configure flexfield data. Otherwise, go to step 5 of this task.

3. In the **Configure Flexfield** window, select an attributes value and click **OK**.

4. See the Flexfields Reference tab for information about the configured flexfield.

5. Enter worker data in the spreadsheet.
   Ensure that you provide any required values and follow instructions in the spreadsheet for creating rows.

Importing Worker Data and Correcting Import Errors

Use the default values except where indicated.

1. In the workers spreadsheet, click **Upload**.

2. In the **Upload Options** window, click **OK**.
   As each row of data is uploaded to the Load Batch Data stage tables, its status is updated.

3. When uploading completes, identify any spreadsheet rows with the status **Insert Failed**, which indicates that the row was not imported to the stage tables.

4. For any row that failed, double-click the status value to display a description of the error.

5. Correct any import errors and click **Upload** again to import the remaining rows to the same batch.
   As rows are imported successfully to the stage tables, the data is loaded automatically to the application tables.

Reviewing and Correcting Load Errors

1. In the spreadsheet, click **Refresh** to display latest load status.
   Any errors that occur during the load process are reported in the spreadsheet.

2. Correct any load errors in the spreadsheet.

3. Repeat this process from Importing Worker Data and Correcting Import Errors until all spreadsheet rows are both imported and loaded successfully.

4. Close the spreadsheet.
   To load a second batch of worker users on the same date, increment the batch number in the spreadsheet and batch names (for example, `Workers042713Batch02`).
Manage Users

Creating Users: Worked Example

You can create users by entering basic person and employment data. A user account is created automatically for a person when you create the user record. You can assign the users Oracle Fusion Human Capital Management (HCM) and non-HCM data roles, each providing access to specific functions and data. This example demonstrates how to create a user and assign roles to the user.

**Note**

This user management functionality is available for HCM Foundation and Oracle Fusion Workforce Directory Management (WDM) users only.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In this Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>For whom are you creating the user record?</td>
<td>Gail Williams</td>
</tr>
<tr>
<td>What is the user account name?</td>
<td>Same as the e-mail ID, <a href="mailto:gail.williams@vision.com">gail.williams@vision.com</a></td>
</tr>
<tr>
<td>Where is Gail employed?</td>
<td>Gail is an employee of Vision Corporation, and works in the Human Resources (HR) department in the Canada office.</td>
</tr>
<tr>
<td>What roles must be provisioned to Gail?</td>
<td>Autoprovision the employee role. Gail is responsible for processing workers’ expense claims so provision the role Expense Claims Administrator manually to Gail.</td>
</tr>
</tbody>
</table>

**Prerequisites**

1. Create a role mapping called All Employees and enter the following conditions.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Assignment Status</td>
<td>Active</td>
</tr>
</tbody>
</table>

In the role mapping you include the:

- Employee role, and select the **Autoprosition** option
- Expense Claims Administrator role, and select the **Self-requestable** option

**Creating a User**

1. On the Search Person page, click the **Create** icon to open the Create User page.
2. Complete the fields, as shown in this table:
3. In the User Details region, leave the User Name field blank. The user name defaults to the user's e-mail ID.

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

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

### Assigning Roles to the User

1. Click **Autoprovision Roles** to provision the employee role to the user.

2. Click **Add Role**.

3. Search for and select the **Expense Claims Administrator** role.

4. Click **Save and Close**. The user account is created and the roles are assigned to the user immediately.

### 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>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.

**CreatedBy**

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 autoprovision roles?**

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

Roles for which the **Autoprovision** 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 autoprovision 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.
Common Applications Configuration: Define Approval Management for Project Execution 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, with the condition that the column name, SECURITY_CODE, is equal to the value HR.
Common Applications Configuration: Maintain Common Reference Objects

Maintain Common Reference Objects: Overview

The Maintain Common Reference Objects task list contains Oracle Middleware Extensions for Applications (Applications Core) tasks that support implementation of common behaviors, such as data security or reference data sets.

Use this task list to manage common reference objects that are defined centrally and shared across applications, in addition to those that are specific to Applications Core functionality. You can access this task list by starting in the Setup and Maintenance Overview page and searching for common reference object task lists.

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

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

Define Application Taxonomy

Application Taxonomy: Highlights

Application taxonomy is the organization of Oracle application components and functions in a hierarchical structure, from product lines to logical business areas. This hierarchy represents a breakdown of products into units based on how applications are installed and supported. Maintain this hierarchy on the Manage Taxonomy Hierarchy page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Taxonomy Hierarchy task.

A detailed introduction to application taxonomy is provided in the Oracle Fusion Applications Developer’s Guide.
Hierarchy

- The application taxonomy hierarchy contains various levels and types of nodes, or modules.

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

Usage

- Use application taxonomy to understand relationships among applications and between an application and its files. This information is helpful in managing various phases of the product lifecycle.

  See: How to Manage the Lifecycle

Modules in Application Taxonomy: Explained

A module is any node in the application taxonomy hierarchy. The top level of the hierarchy is product line, followed by product family, application, and logical business area. There can be multiple levels of logical business areas, with one or more nested within a parent logical business area.

Product Line

A product line is a collection of products under a single brand name, for example, Oracle Fusion.

Product Family

A product family is a collection of products associated with a functional area that may or may not be licensed together as a single unit, for example Financials.

Application

An application is a single product within a product family, containing closely related features for a specific business solution, for example General Ledger.

Logical Business Area

A logical business area is a collection of business object definitions organized into a logical grouping. It contains the model objects, services, and UI components for those business objects. Logical business areas have their own hierarchy levels and in some cases can be two or three levels deep. Each leaf node has at least one business object and service, up to a maximum of four business objects and associated services. A logical business area with more than four business objects are further refined with child logical business area levels. Each of these parent-child levels is represented by a directory in the physical package hierarchy.
Managing Modules in Application Taxonomy: Points to Consider

Manage modules on the Create Child Module or Edit Module page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Taxonomy Hierarchy task. When you create a module, it is a child of the currently selected node in the application taxonomy hierarchy. This determines which values are available, for example for module type. Once created, you cannot delete the module or move it elsewhere in the hierarchy. As you create or edit modules, consider the following points regarding specific fields.

Identifiers

Module ID is the unique primary key for nodes in the taxonomy table. When you create a module, an ID is automatically generated. Once the module is created, you cannot update the ID.

Module key and alternative ID are additional identifiers of the module, presented in a way that is easier to read than the module ID. The module key is a string identifier, for example AP for the Oracle Fusion Payables application. The alternative ID is a numeric identifier, for example 1 for the Oracle Fusion product line. These IDs are provided for the product line, product family, and application modules, but you can optionally add them for logical business areas and new custom modules.

Note

Do not change the module key or alternative ID for predefined modules.

The product code is relevant only to application and logical business area modules. You can leave the field blank for other module types. The product code for applications is the short name that can be displayed in lists of application values, for example FND for Oracle Middleware Extensions for Applications.

Names

Module name is the logical name for the module and is always available. The name must be unique among nodes in the same hierarchy level with the same parent, but try to make it as unique in the whole hierarchy as possible.

The user name and description can appear to users in other parts of Oracle Fusion Applications, so make sure that the values are something that users know to represent the module.

Usage Types

Though you can update the usage type to reflect the current state of the module, just doing so does not affect the actual state. For example, setting a module as installed does not mean it is actually installed if the installation itself has not taken place. Installation refers to operations related to laying down all the components needed to create an Oracle Fusion Applications environment, while
deployment is the process that starts the managed servers and clusters and facilitates the actual use of product offerings. A licensed module is available for installation and deployment, and a deployed module is considered actively used when actually used by users.

Seed Data

If seed data is allowed, then seed data such as flexfields and lookups can be extracted for the module using seed data loaders. By default, extract is allowed for all predefined modules of type application and logical business area.

Associations

You can associate a logical domain to modules of type product family, as well as one or more enterprise applications to modules of type application. This association represents the relationship between the taxonomy modules and the corresponding domain and enterprise applications stored in the Oracle Fusion Applications Functional Core (ASK) tables.

Define Reference Data Sharing

Reference Data Sharing: Explained

Reference data sharing facilitates sharing of configuration data such as jobs and payment terms, across organizational divisions or business units. You define reference data sets and determine how the data is shared or partitioned. Use reference data sets to reduce duplication and maintenance by sharing common data across business entities where appropriate. Depending on the requirement (specific or common), each business unit can maintain its data at a central location, using a set of values either specific to it or shared by other business units.

You can share reference data after it is filtered on the basis of sets. A common reference data set is available as the default set, which can be assigned to several business units sharing the same reference data. For commonly used data such as currencies, you can use the common reference data set and assign it to multiple business units in various countries that use the same currency. In cases where the default set cannot be assigned to an entity, you can create specific sets. The data set visible on the transactional page depends on the sharing method used to share reference data.

For example, XYZ Corporation uses the same grades throughout the entire organization. Instead of managers in different business units setting up the same grades, XYZ Corporation decides to create a set called Grades and assign the grades reference data group for all business units in the organization to the Grades set, so that the grades can be shared.

Note

For specific information on configuring reference data sharing for a particular object or product, refer to its product documentation.
Reference Data Sets: Explained

Reference data sets are logical groups of reference data that can be accessed by various transactional entities depending on the business context. Oracle Fusion Applications contains a common reference data set as well as an enterprise set that may be used as a default set. Depending on your business requirement you can create and maintain additional reference data sets, while continuing to use the common reference data set.

Consider the following scenario.

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

Partitioning

The partitioning of reference data and creation of data sets enable you to create reference entities across tables or lookup types, and share modular information and data processing options among business units. With the help of partitioning, you can choose to create separate sets and subsets for each business unit depending upon its business requirement, or create common sets or subsets to enable sharing reference data between several business units, without the need for duplicating the reference data. Partitioning provides you the flexibility to handle the reference data in a way appropriate to your business needs.

The following figure illustrates the reference data sharing method (assignment to one set only, with common values) where the user can access the data assigned to a specific set in a particular business unit, as well as access the data assigned to the common set.
Reference Data Sets and Sharing Methods: Explained

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

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

Reference Data Sets

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

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

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

Reference Data Sharing Methods

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

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

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

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

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

Assigning Reference Data Sets to Reference Objects: Points to Consider

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

Determinant Types

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

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

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

Reference Groups

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

Note

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

Define ISO Reference Data

Defining Currencies: Points to Consider

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

Currency Codes

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

Date Ranges

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

Symbols

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

**Euro Currency Derivation: Explained**

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

**Note**

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

**Derivation Type**

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

**Derivation Factor**

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

**Derivation Effective Date**

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

**Natural Languages: Points to Consider**

Natural languages are all the languages that humans use, written and spoken. If a language is enabled, then users can associate it with entities, for example as languages spoken by sales representatives. When managing natural languages, consider tasks to perform and best practices for entering particular values.

**Tasks**

Once you add a language, it cannot be deleted, just disabled. You can optionally associate natural languages with International Organization for Standardization (ISO) languages and territories, just for reference.
Values

When you create a natural language, use the alpha-2 ISO code as the language code, or, if not available, then alpha-3. If the language is not an ISO language, then use x- as a prefix for the code, for example x-ja for a Japanese dialect. Use the sgn code of ISO-639-2 for sign languages, followed by territory code, for example sgn-US for American Sign Language. You can also use Internet Assigned Numbers Authority (IANA) language tags.

The natural language description should be the language name with territory name in parenthesis where needed, for example English (Australia) and English (Canada).

FAQs for Define ISO Reference Data

When do I create or edit territories?

Edit territory descriptions to determine how they are displayed in lists of country values throughout Oracle Fusion Applications. The predefined territories are all countries from the International Organization for Standardization (ISO) 3166 standard. You usually would not edit territory names or codes.

Do not edit National Language Support (NLS) territory codes, which are identifiers used in the system, unless you need to change the association between ISO and system territory. You usually would not edit the default currency, which is the value that defaults in the Currency field in Oracle Fusion Applications user preferences after the user first selects a territory.

Create territories if new countries emerge and the system has not yet been patched with the latest ISO country values.

When do I create or edit industries?

Edit industry descriptions to determine how they are displayed in Oracle Fusion Applications. You usually would not edit industry names, which are from the North American Industry Classification System (NAICS). Enabled industries are mainly used in the context of customization, though these values can also appear in any application.

Create industries if you have particular ones you need, for example for customization, that are not included in the NAICS standard.

When do I associate industries with territories?

Optionally associate industries with territories to provide an industry in territory value, used for customization. For example, administrators can customize a page in one way for users within an industry in one country, and another way for users within the same industry in another country. The administrator would select the appropriate industry in territory value to set the customization context.
**When do I create or enable currencies?**

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

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

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

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

**Note**

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

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

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

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

**When do I create or edit ISO languages?**

You can edit the names and descriptions of International Organization for Standardization (ISO) languages to determine how they are displayed in lists of ISO language values in Oracle Fusion Applications. The ISO languages are from the ISO 639 standard. If there were changes to the ISO standard and the system has not yet been patched with the latest ISO values, you can update the ISO alpha-2 code or add languages as needed.

**When do I edit languages?**

Installed languages automatically appear on the Manage Languages page, so you do not manually enter newly installed languages. This page contains
all languages available for installation and translation in Oracle Fusion Applications. Each dialect is treated as a separate language. The language codes and names are values used by the system.

You generally would not edit any of the detailed fields unless you really need to and know what they are.

**When do I create or edit time zones?**

Though all standard time zones are provided, optionally enable only a subset for use in lists of time zone values in Oracle Fusion Applications. You can add time zones if new zones became standard and the system has not yet been patched with the latest values.

**Manage Data Security Policies**

**Data Security in the Security Reference Implementation: Explained**

The reference implementation contains a set of data security policies that can be inspected and confirmed to be suitable or a basis for further implementation using the Authorization Policy Manager (APM).

The security implementation of an enterprise is likely a subset of the reference implementation, with the enterprise specifics of duty roles, data security policies, and HCM security profiles provided by the enterprise.

The business objects registered as secure in the reference implementation are database tables and views.

Granting or revoking object entitlement to a particular user or group of users on an object instance or set of instances extends the base Oracle Fusion Applications security reference implementation without requiring customization of the applications that access the data.

**Data Security Policies in the Security Reference Implementation**

The data security policies in the reference implementation entitle the grantee (a role) to access instance sets of data based on SQL predicates in a WHERE clause.

---

**Tip**

When extending the reference implementation with additional data security policies, identify instance sets of data representing the business objects that need to be secured, rather than specific instances or all instances of the business objects.

---

Predefined data security policies are stored in the data security policy store, managed in the Authorization Policy Manager (APM), and described in the Oracle Fusion Applications Security Reference Manual for each offering. A data
security policy for a duty role describes an entitlement granted to any job role that includes that duty role.

**Warning**

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

The reference implementation only enforces a portion of the data security policies in business intelligence that is considered most critical to risk management without negatively affecting performance. For performance reasons it is not practical to secure every level in every dimension. Your enterprise may have a different risk tolerance than assumed by the security reference implementation.

**HCM Security Profiles in the Security Reference Implementation**

The security reference implementation includes some predefined HCM security profiles for initial usability. For example, a predefined HCM security profile allows line managers to see the people that report to them.

The IT security manager uses HCM security profiles to define the sets of HCM data that can be accessed by the roles that are provisioned to users.

**Data Roles**

The security reference implementation includes no predefined data roles to ensure a fully secured initial Oracle Fusion Applications environment.

The security reference implementation includes data role templates that you can use to generate a set of data roles with entitlement to perform predefined business functions within data dimensions such as business unit. Oracle Fusion Payables invoicing and expense management are examples of predefined business functions. Accounts Payable Manager - US is a data role you might generate from a predefined data role template for payables invoicing if you set up a business unit called US.

HCM provides a mechanism for generating HCM related data roles.

**Data Security: Explained**

By default, users are denied access to all data.

Data security makes data available to users by the following means.

- Policies that define grants available through provisioned roles
- Policies defined in application code

You secure data by provisioning roles that provide the necessary access. Enterprise roles provide access to data through data security policies defined for the inherited application roles.
When setting up the enterprise with structures such as business units, data roles are automatically generated that inherit job roles based on data role templates. Data roles also can be generated based on HCM security profiles. Data role templates and HCM security profiles enable defining the instance sets specified in data security policies.

When you provision a job role to a user, the job role implicitly limits data access based on the data security policies of the inherited duty roles. When you provision a data role to a user, the data role explicitly limits the data access of the inherited job role to a dimension of data.

Data security consists of privileges conditionally granted to a role and used to control access to the data. A privilege is a single, real world action on a single business object. A data security policy is a grant of a set of privileges to a principal on an object or attribute group for a given condition. A grant authorizes a role, the grantee, to actions on a set of database resources. A database resource is an object, object instance, or object instance set. An entitlement is one or more allowable actions applied to a set of database resources.

Data is secured by the following means.

<table>
<thead>
<tr>
<th>Data security feature</th>
<th>Does what?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data security policy</td>
<td>Grants access to roles by means of entitlement</td>
</tr>
<tr>
<td>Role</td>
<td>Applies data security policies with conditions to users through role provisioning.</td>
</tr>
<tr>
<td>Data role template</td>
<td>Defines the data roles generated based on enterprise setup of data dimensions such as business unit.</td>
</tr>
<tr>
<td>HCM security profile</td>
<td>Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code.</td>
</tr>
<tr>
<td>Masking</td>
<td>Hides private data on non-production database instances</td>
</tr>
<tr>
<td>Encryption</td>
<td>Scrambles data to prevent users without decryption authorization from reading secured data</td>
</tr>
</tbody>
</table>

The sets of data that a user can access via roles are defined in Oracle Fusion Data Security. Oracle Fusion Data Security integrates with Oracle Platform Security Services (OPSS) to entitle users or roles (which are stored externally) with access to data. Users are granted access through the entitlement assigned to the roles or role hierarchy with which the user is provisioned. Conditions are WHERE clauses that specify access within a particular dimension, such as by business unit to which the user is authorized.

**Data Security Policies**

Data security policies articulate the security requirement "Who can do What on Which set of data," where 'Which set of data' is an entire object or an object instance or object instance set and 'What' is the object entitlement.

For example, accounts payable managers can view AP disbursements for their business unit.
A data security policy is a statement in a natural language, such as English, that typically defines the grant by which a role secures business objects. The grant records the following.

- Table or view
- Entitlement (actions expressed by privileges)
- Instance set (data identified by the condition)

For example, disbursement is a business object that an accounts payable manager can manage by payment function for any employee expenses in the payment process.

**Note**

Some data security policies are not defined as grants but directly in applications code. The security reference manuals for Oracle Fusion Applications offerings differentiate between data security policies that define a grant and data security policies defined in Oracle Fusion applications code.

A business object participating in a data security policy is the database resource of the policy.

Data security policies that use job or duty roles refer to data security entitlement. For example, the data security policy for the Accounts Payable Manager job role refers to the view action on AP disbursements as the data security entitlement.

**Important**

The duty roles inherited by the job role can be moved and job roles reassembled without having to modify the data security.

As a security guideline, data security policies based on user session context should entitle a duty role. This keeps both function and data security policies at the duty role level, thus reducing errors.

For example, a Sales Party Management Duty can update Sales Party where the provisioned user is a member of the territory associated with the sales account. Or the Sales Party Management Duty can update Sales Party where the provisioned user is in the management chain of a resource who is on the sales account team with edit access. Or the Participant Interaction Management Duty can view an Interaction where the provisioned user is a participant of the Interaction.

For example, the Disbursement Process Management Duty role includes entitlement to build documents payable into payments. The Accounts Payable Manager job role inherits the Disbursement Process Management Duty role. Data security policies for the Disbursement Process Management Duty role authorize access to data associated with business objects such as AP disbursements within
a business unit. As a result, the user provisioned with the Accounts Payable Manager job role is authorized to view AP disbursements within their business unit.

A data security policy identifies the entitlement (the actions that can be made on logical business objects or dashboards), the roles that can perform those actions, and the conditions that limit access. Conditions are readable WHERE clauses. The WHERE clause is defined in the data as an instance set and this is then referenced on a grant that also records the table name and required entitlement.

**Data Roles**

Data roles are implemented as job roles for a defined set of data.

A data role defines a dimension of data within which a job is performed. The data role inherits the job role that describes the job. For example, a data role entitles a user to perform a job in a business unit.

The data role inherits abstract or job roles and is granted data security privileges. Data roles carry the function security privileges inherited from job roles and also the data security privilege granted on database objects and table rows.

For example, an accounts payables specialist in the US Business Unit may be assigned the data role Accounts Payables Specialist - US Business Unit. This data role inherits the job role Accounts Payables Specialist and grants access to transactions in the US Business Unit.

A data role may be granted entitlement over a set people.

For example, a Benefits Administrator A-E is allowed to administer benefits for all people that have a surname that begins with A-E.

Data roles are created using data role templates. You create and maintain data roles in the Authorization Policy Manager (APM). Use the Manage Data Roles and Security Profiles task to create and maintain HCM data roles in Oracle Fusion HCM.

**HCM Security Profiles**

HCM security profiles are used to secure HCM data, such as people and departments. You use HCM security profiles to generate grants for an enterprise role. The resulting data role with its role hierarchy and grants operates in the same way as any other data role.

For example, an HCM security profile identifies all employees in the Finance division.

Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

**Masking and Encryption**

Oracle Fusion Applications uses masking to protect sensitive data from view by unauthorized users. Encryption APIs mask sensitive fields in applications user interfaces. Additionally, Oracle Data Masking is available for masking data in non-production instances and Oracle Transparent Data Encryption is available.
for protecting data in transit or in backups independent of managing encryption keys.

**Database Resources and Data Security Policies: How They Work Together**

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

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

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

**Database Resources**

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

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

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

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

Note

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

Actions

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

Conditions

A condition is either a SQL predicate or an XML filter. A condition expresses the values in the data object by a search operator or a relationship in a tree hierarchy. A SQL predicate, unlike an XML filter, is entered in a text field in the data security user interface pages and supports more complex filtering than an XML filter, such as nesting of conditions or sub queries. An XML filter, unlike a SQL predicate, is assembled from choices in the UI pages as an AND statement.

Tip

An XML filter can be effective in downstream processes such as business intelligence metrics. A SQL predicate cannot be used in downstream metrics.

Securing Data Access: Points to Consider

Oracle Fusion Applications supports securing data through role-based access control (RBAC) by the following methods.

<table>
<thead>
<tr>
<th>Method of securing data</th>
<th>Reason</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data roles apply explicit data security policies on job and abstract roles</td>
<td>Appropriate for job and abstract roles that should only access a subset of data, as defined by the data role template that generates the data role or by HCM security profiles.</td>
<td>Accounts Payable Manager - US data role to provide an accounts payable manager in the US business unit with access to invoices in the US business unit.</td>
</tr>
<tr>
<td>Data security policies</td>
<td>Define data access for application roles and provide inheriting job and abstract roles with implicit data security</td>
<td>Projects</td>
</tr>
</tbody>
</table>
If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

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

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

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

**Grants**

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

- Rows that are returned by a given query based on the intended business operation
- Actions that are available for a given row

Grants control which data a user can access.

**Note**

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

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

**Securing a Business Object**

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

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

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

Data Role Templates: Explained

You use data role templates to generate data roles. You generate such data roles, and create and maintain data role templates in the Authorization Policy Manager (APM).

Note

HCM data roles are generated using the Manage Data Roles and Security Profiles task, which uses HCM security profiles, not data role templates, to define the data security condition.

The following attributes define a data role template.

- Template name
- Template description
- Template group ID
- Base roles
- Data dimension
- Data role naming rule
- Data security policies

The data role template specifies which base roles to combine with which dimension values for a set of data security policies. The base roles are the parent job or abstract roles of the data roles.

Note

Abstract, job, and data roles are enterprise roles in Oracle Fusion Applications. Oracle Fusion Middleware products such as Oracle Identity Manager (OIM) and Authorization Policy Manager (APM) refer to enterprise roles as external roles. Duty roles are implemented as application roles in APM and scoped to individual Oracle Fusion Applications.

The dimension expresses stripes of data, such as territorial or geographic information you use to partition enterprise data. For example, business units are a type of dimension, and the values picked up for that dimension by the data role template as it creates data roles are the business units defined for your enterprise. The data role template constrains the generated data roles with grants of entitlement to access specific data resources with particular actions. The data
role provides provisioned users with access to a dimensional subset of the data granted by a data security policy.

An example of a dimension is a business unit. An example of a dimension value is a specific business unit defined in your enterprise, such as US. An example of a data security policy is a grant to access a business object such as an invoice with a view entitlement.

When you generate data roles, the template applies the values of the dimension and participant data security policies to the group of base roles.

The template generates the data roles using a naming convention specified by the template’s naming rule. The generated data roles are stored in the Lightweight Directory Access Protocol (LDAP) store. Once a data role is generated, you provision it to users. A user provisioned with a data role is granted permission to access the data defined by the dimension and data security grant policies of the data role template.

For example, a data role template contains an Accounts Payable Specialist role and an Accounts Payable Manager role as its base roles, and region as its dimension, with the dimension values US and UK. The naming convention is [base-role-name]:[DIMENSION-CODE-NAME]. This data role template generates four data roles.

- Accounts Payable Specialist - US (business unit)
- Accounts Payable Specialist - UK (business unit)
- Accounts Payable Manager - US (business unit)
- Accounts Payable Manager - UK (business unit)

Making Changes To Data Role Templates

If you add a base role to an existing data role template, you can generate a new set of data roles. If the naming rule is unchanged, existing data roles are overwritten.

If you remove a base role from a data role template and regenerate data roles, a resulting invalid role list gives you the option to delete or disable the data roles that would be changed by that removal.

Making Changes to Dimension Values

If you add a dimension value to your enterprise that is used by a data role template, you must regenerate roles from that data role template to create a data role for the new dimension. For example if you add a business unit to your enterprise, you must regenerate data roles from the data role templates that include business unit as a dimension.

If you add or remove a dimension value from your enterprise that is used to generate data roles, regenerating the set of data roles adds or removes the data roles for those dimension values. If your enterprise has scheduled regeneration as an Oracle Enterprise Scheduler Services process, the changes are made automatically.
For information on working with data role templates, see the Oracle Fusion Middleware Administrator’s Guide for Authorization Policy Manager.

### Set Activity Stream Options

#### 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.
### Permitted Tasks

- Add and delete custom items.
- Edit any item.
- Specify navigation to a UI Shell page in an Oracle Fusion application.
- Hide or show items.

### Restricted Tasks

- You cannot add menu items (links) as sub-nodes. All nodes are top-level nodes.
- You cannot delete nodes that are delivered with the product. Instead, you can hide them.
- You cannot move nodes. Instead, you must duplicate the node and hide the original node.

### 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/adf-config.xml file.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

**Working with Nodes**

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

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

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

A node that appears in italics either contains an EL command or the Rendered check box beside it was cleared, and therefore is hidden from end users.
Tip
For major changes that need to be tested and approved, you might want to use the sandbox manager instead of hiding and showing nodes.

Design Time Menu Customizations: Highlights

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

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

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

Customizations

- Use Oracle JDeveloper to customize the Navigator and home page menus at design time.
  See: Customizing Menus
- Define translations for your customizations in the locales you support.
  See: Translating Menu Customizations
- Customize the page template to display the Navigator menu groups as separate menus, each of them displaying their list of menu items. Refer to the Oracle Fusion Applications Developer's Guide.
  See: Rendering the Navigator Menu as Dropdown Buttons

Troubleshooting Navigator Menu: Highlights

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

Issues and Resolutions

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

Managing Audit Policies: Explained

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

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

Enabling Audit Functionality

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

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

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

Configuring Audit Business Object Attributes: Points to Consider

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

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

**Selecting an Application**

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

**Selecting Attributes**

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

**Note**

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

**Starting and Stopping Audit**

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

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

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

**Configuring Audit: Highlights**

You can set up auditing for Oracle Fusion Applications using the Manage Audit Policies page in the Setup and Maintenance work area of Oracle Fusion Applications.
To set up auditing for Oracle Fusion Middleware products, you must select the level of auditing that maps to a predefined set of metadata and events that have to be audited. Information on configuring audit for Oracle Fusion Middleware products is provided in Oracle Fusion Middleware guides.

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

**Oracle Fusion Middleware Products**


  See: Auditing Web Services

**Oracle Fusion Security Products**

- Configure business objects to enable auditing in Oracle Fusion security products. Refer to Oracle Fusion Middleware Application Security Guide.

  See: Oracle Fusion Middleware Audit Framework Reference

**Manage Oracle Social Network Objects**

**Managing Oracle Social Network Objects: Explained**

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

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

- The business objects and attributes to enable

- The enablement method for social network integration with Oracle Fusion Applications

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

Use social network to:

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

Note

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

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

Enabling Business Objects for Integration

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

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

Options for Enabling Oracle Social Network Objects: Explained

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

In the Manage Oracle Social Network Objects page, you can:

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

To access the Manage Oracle Social Network Objects page:

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

Note

7-32 Oracle Project Portfolio Management Cloud Implementing Project Execution Management
Custom objects and attributes created in Application Composer in the mainline are also displayed on the **Manage Oracle Social Network Objects** page. You can enable these objects and attributes for social network integration.

### Enable Object

To enable a business object:

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

2. In the **Business Objects** section, select a business object, click **Enable Object**, and select one of the enablement options. The business objects are grouped by modules. The available enablement options are:
   - **Manual**: (Recommended) Empowers the user to decide whether to share each instance of the object with social network.
   - **Automatic**: Automatically sends the newly enabled object instances and updates to social network.
   - **No**: Does not send any information on object instance to social network. This is the default option.

3. Click **OK**.

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

### 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**.

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.
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**

**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.

**Manage Applications Core 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.

**Manage Applications Core Standard Lookups**

**Lookups: Explained**

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

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

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

<table>
<thead>
<tr>
<th>Lookup Type</th>
<th>Lookup Code</th>
<th>Meaning</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAR_STATUS</td>
<td>M</td>
<td>Married</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Single</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>Registered Partner</td>
<td>+NL</td>
</tr>
<tr>
<td></td>
<td>DP</td>
<td>Domestic Partner</td>
<td>-FR, AU</td>
</tr>
</tbody>
</table>

In this case, tags are used for localizing the codes. All legislations list Married
and Single. Only the Dutch legislation lists Registered Partner. And all
legislations except France and Australia also list Domestic Partner.

When managing lookups, you need to understand the following.

- Using lookups in applications
- Customization levels
- Accessing lookups
- Enabling lookups
- The three kinds of lookups: standard, common, and set enabled

**Using Lookups in Applications**

Use lookups to provide validation or a list of values for a user input field in a
user interface.

An example of a lookup used for validation is a flexfield segment using a
table-validated value set with values from a lookup type. An example of a
lookup in a list of values is a profile option's available values from which users
select one to set the profile option. Invoice Approval Status gives the option of
including payables invoices of different approval statuses in a report. The lookup
code values include All so that users can report by all statuses: Approved,
Resubmitted for approval, Pending or rejected, and Rejected.

**Customization Level**

The customization level of a lookup type determines whether the lookups in that
lookup type can be edited. This applies data security to lookups.

Some lookup types are locked so no new codes and other changes can be added
during implementation or later, as needed. Depending on the customization
level of a lookup type, you may be able to change the codes or their meanings.
Some lookups are designated as extensible, so new lookup codes can be created
during implementation, but the meanings of predefined lookup codes cannot be
modified. Some predefined lookup codes can be changed during implementation
or later, as needed.
The customization levels are user, extensible, and system. The following table shows which lookup management tasks are allowed at each customization level.

<table>
<thead>
<tr>
<th>Allowed Task</th>
<th>User</th>
<th>Extensible</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deleting a lookup type</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Inserting new codes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Updating start date, end date, and</td>
<td>Yes</td>
<td>Yes, only</td>
<td>No</td>
</tr>
<tr>
<td>enabled fields</td>
<td></td>
<td>if the code is not predefined data</td>
<td></td>
</tr>
<tr>
<td>Deleting codes</td>
<td>Yes</td>
<td>Yes, only</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td></td>
<td>if the code is not predefined data</td>
<td></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>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>
</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 Manage Applications Core Standard Lookups

How can I edit lookups?

You can edit the existing lookup codes of a lookup type or add new lookup codes on the Define Lookups pages, which you can access by starting in the Setup and Maintenance work area and searching for lookup tasks. You can edit the existing lookup codes of a lookup type, or add new lookup codes to a lookup type, if the customization level for the lookup type supports editing.

Why can't I see my lookup types?

Lookups are listed by lookup type. Typically lookup types are managed using tasks that handle a group of related lookups, such as Manage Geography Lookups. Each task gives you access only to certain lookup types. The generic tasks provide access to all lookup types of a kind, such as all common lookups using the Manage Common Lookups task.

If existing lookups are not available to the tasks of the Define Lookups activity, they may be validated for use in a lookup view that is not central to all applications or whose owning application has not been specified in a lookup view.

Lookups can only be managed in the Define Lookups tasks if the lookup’s view application is the standard lookups view, common lookups view, or set-enabled lookups view. Lookups defined in an application view can only be managed by following instructions provided by the owning application.

Note

A lookup type and its codes can only be defined in one lookup view.

What's the difference between a lookup type and a value set?

A lookup type consists of lookup codes that are the values in a static list of values. Lookup code validation is a one to one match.

A table-validated value set can consist of values that are validated through a SQL statement, which allows the list of values to be dynamic.
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 Applications Core Profile Options**

**Profile Options: Explained**

Profile options manage configuration data centrally and influence the behavior of applications. Profile options serve as permanent user preferences and
application configuration parameters. You configure profile options with settings for specific contexts or groups of users. Users customize how their user interfaces look and behave by changing the values of available profile options.

Profile options store the following kinds of information.

<table>
<thead>
<tr>
<th>Type of Information</th>
<th>Profile Option Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>User preferences</td>
<td>Settings to provide access to social networking features</td>
</tr>
<tr>
<td>Installation information</td>
<td>Setting to identify the location of a portal</td>
</tr>
<tr>
<td>Configuration choices</td>
<td>Settings to change user interface skins and behaviors</td>
</tr>
<tr>
<td>Processing options</td>
<td>Settings to affect how much information to log either for an entire site or a specific user</td>
</tr>
</tbody>
</table>

You can add and configure new profile options in addition to configuring predefined profile options that are implemented as updateable.

**Profile Option Definition and Configuration**

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

Profile option definitions consist of the following.

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

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

**Profile Option Values**

Some profile options have predefined profile option values.

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

You can set profile option values at different levels: site, product, and user. The following table provides examples.
<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.

```sql
SELECT DESCRIPTION Language, NLS_LANGUAGE FROM FND_LANGUAGES_VL WHERE INSTALLED_FLAG IN ('B','I') ORDER BY DESCRIPTION
```

This results in the following list of values based on data in FND_LANUGUAGE_VL.

<table>
<thead>
<tr>
<th>Display Value</th>
<th>Hidden Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>American English</td>
<td>US</td>
</tr>
<tr>
<td>French</td>
<td>F</td>
</tr>
<tr>
<td>Spanish</td>
<td>E</td>
</tr>
</tbody>
</table>

Hidden values must be varchar2(2000).

Profile options generally provide configuration values within a particular context. Though you can create a profile option to be global, think of global values as default values to avoid storing inappropriate configuration information as profile option values. Create global profile options that have corresponding contextual levels.

**Managing Profile Option Categories: Points to Consider**

Use profile option categories to group profile options.
Organizing Profile Options in Categories

As a guideline, group profile options in a single category if the profile options affect the same feature, or if an administrator would likely want to see the profile options in the results of a single search.

Application developers are responsible for the initial groupings and then administrators can make changes based on their specific needs. Administrators can categorize profile options and then easily search on profile options by category.

Tip

Define profile option categories first and assign new profile options to existing categories rather than defining profile options first and then defining categories to categorize them.

Adding New Profile Option Categories

You can add new categories or add profiles to an existing category.

You can create a profile option category by duplicating an existing category and editing it for a new grouping of profile options. You can add multiple profile options to a category. A profile option can exist in multiple categories.

Profile Option Order in a Category

Specify a profile option sequence to determine the order of profile options when queried by profile option category.

Viewing and Editing Profile Option Values: Points to Consider

A profile option value consists of the value and the context or level where the value is set. You specify the context with a pairing of the profile option value’s level and level value, such as the product level and the level value GL for Oracle Fusion General Ledger. Adding or modifying profile option values can include deciding which valid values are enabled or updatable at which level.

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

Profile Option Levels and User Session Context

Site level profile option values affect the way all applications run for a given implementation. Product level profile option values affect the way applications owned by a particular product code behave. For example, a product may use profile options set at the product level to determine how regions provided by a common module such as those available from Oracle Fusion Trading Community Model or Customer Relationship Management (CRM) display in...
a particular work area or dashboard. User level profile option values affect the way applications run for a specific application user.

Whichever profile option value is most specific to a user session, that is the value at which the profile option is set for the user session.

For example, the predefined FND_LANGUAGE profile option sets the default language. In addition to a site level value, you can define a value for various product or user levels.

<table>
<thead>
<tr>
<th>Level Name</th>
<th>Level Value</th>
<th>Profile Option Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site</td>
<td>InFusion</td>
<td>American English</td>
</tr>
<tr>
<td>Product</td>
<td>Customer Center</td>
<td>French</td>
</tr>
<tr>
<td>Product</td>
<td>CRM Application Composer</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Application Administrator</td>
<td>American English</td>
</tr>
<tr>
<td>User</td>
<td>Hima</td>
<td>Hindi</td>
</tr>
</tbody>
</table>

Values at the site level take effect for any user unless overridden by a different value set at the more specific levels of product and user. Product level profile option values affect the way applications owned by a particular product code behave. In addition to user level profile option values in applications, selections may be available in the user preferences workspace.

The following table demonstrates the FND_LANGUAGE profile option settings that would apply to specific users, based on the example above. For example, the user Hima is using the CRM Application Composer product, in the InFusion site. The example above shows that this profile option is set to Hindi at the user level for Hima. Because user is the highest applicable level for Hima, the applicable profile option value is Hindi for Hima.

<table>
<thead>
<tr>
<th>Site</th>
<th>Product</th>
<th>User</th>
<th>Highest Available Level</th>
<th>Active Profile Option Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>InFusion</td>
<td>CRM Application Composer</td>
<td>Hima</td>
<td>User</td>
<td>Hindi</td>
</tr>
<tr>
<td>Acme</td>
<td>Payables</td>
<td>Application Administrator</td>
<td>User</td>
<td>American English</td>
</tr>
<tr>
<td>InFusion</td>
<td>Customer Center</td>
<td>Guillaume</td>
<td>Product</td>
<td>French</td>
</tr>
<tr>
<td>InFusion</td>
<td>Payables</td>
<td>Implementation Consultant</td>
<td>Site</td>
<td>American English</td>
</tr>
<tr>
<td>Acme</td>
<td>Payables</td>
<td>Implementation Consultant</td>
<td>none</td>
<td>no value</td>
</tr>
</tbody>
</table>

**Note**

More than one site level value is relevant in an enterprise with multiple tenants using a single instance of Oracle Fusion Applications.

**Effect of Changes to Profile Option Values**

Any change you make to a user level profile option has an immediate effect on the way applications run for that session. When you sign in again, changes made
to your user level profile options in a previous session are still in effect. When you change profile option value at the product level and no user level values are set, you see the update immediately, but other users may not see the changed value until signing out and back in. When you change a profile option value and the new value affects other users, the change takes effect only when users sign in the next time.

Changes to site level profile options take effect for any user session that is started after the setting has been changed. Changes to site or user level profile options do not affect any user sessions that are already in progress when the change is made.

Changes to site or user level profile options take effect for any C or PL/SQL processes, such as scheduled jobs, that are launched after the setting has been changed. Profile option changes do not affect C or PL/SQL processes that are already running.

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.
Manage Applications Core 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 runtime page where the flexfield appears.

The following aspects are important in defining value sets:

- Value sets for context segments
- Format-only validation
- Interdependent value sets
- Table validation
- Range
• Security
• Testing and maintenance

Value Sets for Context Segments

When assigning a value set to a context segment, you can only use table-validated or independent value sets.

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being stored must not be larger than the context’s column length. If you use a table value set, the value set cannot reference flexfield segments in the value set’s WHERE clause other than the flexfield segment to which the value set is assigned.

Format Only Validation

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

For example, if the value set allows only numeric characters, users can enter the value 456 (for a value set with maximum length of three or more), but can’t enter the value ABC. A format only value set doesn’t otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how 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</td>
</tr>
<tr>
<td></td>
<td>Dependent, Independent, and Format validation types.</td>
</tr>
<tr>
<td>Maximum length</td>
<td>Maximum number of characters or digits for Character data type.</td>
</tr>
<tr>
<td>Precision</td>
<td>Maximum number of digits the user can enter.</td>
</tr>
<tr>
<td>Scale</td>
<td>Maximum number of digits that can follow the decimal point.</td>
</tr>
<tr>
<td>Uppercase only</td>
<td>Lowercase characters automatically changed to uppercase.</td>
</tr>
<tr>
<td>Zero fill</td>
<td>Automatic right-justification and zero-filling of entered numbers (affects</td>
</tr>
<tr>
<td></td>
<td>values that include only the digits 0-9).</td>
</tr>
</tbody>
</table>

Caution

You cannot change the text value data type to a translated text value subtype after creating a value set. If there is any chance you may need to translate displayed values into other languages, choose Translated text. Selecting the Translated text subtype doesn't require you to provide translated values.

Value Sets for Context Segments

You can use only table and independent value sets to validate context values. The data type must be character and the maximum length of the values being
stored must not be larger than the context's column length. If you use a table value set, the value set cannot reference flexfield segments in the value set's WHERE clause other than the flexfield segment to which the value set is assigned.

Security

When enabling security on a value set, the data security resource name is an existing value set or one that you want to create. The name typically matches the code value for the value set.

Restriction

You cannot edit the data security resource name after you save your changes.

Table-Validated Value Sets and Bind Variables: Points to Consider

After you assign a value set to a flexfield, you can use bind variables in the WHERE clause.

The following bind variables refer to flexfield elements:

- :{SEGMENT.<segment_code>}
- :{CONTEXT.<context_code>;SEGMENT.<segment_code>}
- :{VALUESET.<value_set_code>}
- :{FLEXFIELD.<internal_code>}
- :{PARAMETER.<parameter_code>}

Segment Code

:{SEGMENT.<segment_code>}

This bind variable refers to the ID or value of a segment where the segment identifies the segment. Where referring to the ID, the value set is ID-validated. Where referring to the value, the value set isn't ID-validated. The data type of the bind value is the same as the data type of the segment's column.

For both descriptive and extensible flexfields, the segment must be in the same context as the source segment. The source segment contains the WHERE clause. For descriptive flexfields, if the segment is global, then the source segment must be global.

The segment must have a sequence number that is less than the sequence number of the target segment with this bind variable. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment. For example, the values to select from a CITIES table might depend upon the selected country. If SEGMENT1 contains the country value, then the WHERE clause for the CITIES table might be <country_code> = :{SEGMENT.SEGMENT1}.

Context Code

::{CONTEXT.<context_code>;SEGMENT.<segment_code>}
This bind variable, which is valid only for extensible flexfields, refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of a segment that is in a different context than the target segment (the segment with the WHERE clause).

- The `<context_code>` identifies the context and must be in the same category or in an ancestor category. It cannot be a multiple-row context.
- The `<segment_code>` identifies the segment. The data type of the bind value is the same as the data type of the segment's column.

**Tip**

The target segment should appear in the UI after the source segment to ensure the source segment has a value. If the target segment's context is a single-row context, the source and target segments must be on separate pages and the target page must follow the source page.

This bind variable is useful when the set of valid values depends on the value of a segment in another context. For example, the values to select from a CERTIFICATION table for a segment in the Compliance and Certification context might depend on the value of the country segment in the Manufacturing context.

**Value Set Code**

`:VALUESET.<value_set_code>`

This bind variable refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of the segment that is assigned to the value set that is identified by the `<value_set_code>`. The data type of the bind value is the same as the data type of the segment's column.

The segment must have a sequence number that is less than the sequence number of the segment with this bind variable. If more than one segment is assigned to the value set, the closest prior matching segment will be used to resolve the bind expression. A matching segment must exist in the current flexfield context.

This bind variable is useful when the set of valid values depends on the value in another segment and that segment code can vary, such as when the value set is used for more than one context or flexfield. For example, the values to select from a CITIES table might depend upon the selected country. If the value set for the segment that contains the country value is COUNTRIES, then the WHERE clause for the CITIES table might be `<county_code> = :{VALUESET.COUNTRIES}`.

**Flexfield Internal Code**

`:FLEXFIELD.<internal_code>`

This bind variable refers to an internal code of the flexfield in which the value set is used, or to a validation date. The internal_code must be one of the following:

- APPLICATION_ID - the application ID of the flexfield in which this value set is used. The data type of APPLICATION_ID and its resulting bind value is NUMBER.
- DESCRIPTIVE_FLEXFIELD_CODE - the identifying code of the flexfield in which this value set is used. The data type of
DESCRIPTIVE_FLEXFIELD_CODE and its resulting bind value is VARCHAR2. Note that you use this string for both descriptive and extensible flexfields.

- CONTEXT_CODE - the context code of the flexfield context in which this value set is used. The data type of CONTEXT_CODE and its resulting bind value is VARCHAR2.

- SEGMENT_CODE - the identifying code of the flexfield segment in which this value set is used. The data type of SEGMENT_CODE and its resulting bind value is VARCHAR2.

- VALIDATION_DATE - the current database date. The data type of VALIDATION_DATE and its resulting bind value is DATE.

**Flexfield Parameters**

`:( PARAMETER.<parameter_code>)`

This bind variable refers to the value of a flexfield parameter where parameter_code identifies the parameter. The data type of the resulting bind value is the same as the parameter's data type.

**Note**

You cannot assign a table value set to a context segment if the WHERE clause uses VALUESET.value_set_code or SEGMENT.segment_code bind variables.

---

**Table-Validated Value Set: Worked Example**

In an application user interface, you want to display a list of values that allow customers to enter satisfaction scores. The value column name is 1, 2, 3, 4, 5 and the value column description is Extremely Satisfied, Satisfied, and so on. Users can pick the appropriate value or description which stores the corresponding name so the name value can be used in a calculation expression.

In this case, you can use the FND_LOOKUPS table as the basis for a table-validated value set. The lookup meaning corresponds to the Value Column Name and the lookup description corresponds to the Description Column Name. The properties of the value set are as follows:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM clause</td>
<td>FND_LOOKUPS</td>
</tr>
<tr>
<td>WHERE clause</td>
<td><code>lookup_type = 'CN_XX_CUST_SATISFACT_SCORE'</code></td>
</tr>
<tr>
<td>ID column</td>
<td><code>lookup_code</code></td>
</tr>
<tr>
<td>Value column</td>
<td><code>meaning</code></td>
</tr>
<tr>
<td>Description column</td>
<td><code>description</code></td>
</tr>
<tr>
<td>Enable Flag column</td>
<td><code>enabled_flag</code></td>
</tr>
<tr>
<td>Start Date column</td>
<td><code>start_date_active</code></td>
</tr>
<tr>
<td>End Date column</td>
<td><code>end_date_active</code></td>
</tr>
<tr>
<td>Order by</td>
<td><code>display_sequence</code></td>
</tr>
</tbody>
</table>
After completing this task, you should have created your customer satisfaction value set for the Incentive Compensation page of your implementation project.

**Creating a Value Set Based on a Lookup**

1. From the Setup and Maintenance work area, find the Manage Value Sets task and click the **Go to Task** icon button.

2. On the Manage Value Sets page, click the **Create** icon button.

3. On the Create Value Set page, enter the following values:
   a. In the Value Set Code field, enter `CN_XX_CUSTOMER_SATISFACTION_SCORES`
   b. In the Description field, enter Customer satisfaction score.
   c. In the Module field, select **Search**.
   d. In the Search and Select: Module subwindow, enter `Incent` in the User Module Name field
   e. Select Incentive Compensation.
   f. Click **OK**.

4. On the Create Value Set page, enter the following values:
   a. In the Validation Type field, select Table.
   b. In the Value Data Type field, select Character.
   c. In the Definition section FROM Clause field, enter `FND_LOOKUPS`.
   d. In the Value Column Name field, enter DESCRIPTION.
   e. In the Description Column Name field, enter MEANING.
   f. In the ID Column Name field, enter LOOKUP_CODE.
   g. In the Enabled Flag Column Name field, enter 'Y'.
   h. In the Start Date Column Name field, enter START_DATE_ACTIVE.
   i. In the End Date Column Name field, enter END_DATE_ACTIVE.
   j. In the WHERE Clause field, enter `LOOKUP_TYPE = 'CN_XX_CUST_SATISFACT_SCORE'`.

5. Click **Save and Close**.

6. In the Manage Value Sets page, click **Done**.

**Adding Attributes to the Manage Value Sets Page: 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.

**Translating Flexfield and Value Set Configurations: Explained**

When you first configure a flexfield or segment, the translatable text that you enter, such as prompts and descriptions, is stored as the text for all installed locales. You may then provide a translation for a particular locale. If you don’t provide a translation for a given locale, then the value that was first entered is used for that locale.

To translate the text for a particular locale, log in with that locale or specify the locale by selecting **Settings and Actions - Personalization - Set Preferences** in the global area. Then, update the translatable text in the flexfield using the Manage Descriptive Flexfields task, Manage Key Flexfields task, or Manage Extensible Flexfields task. Your modifications change the translated values only for the current session’s locale.

After you complete the translations, deploy the flexfield.

You can define translations for a dependent value set or an independent value set, if it is of type Character with a subtype of Translated text. You define the translations by setting the current session to the locale for which you want to define the translation and using the Manage Value Sets task to enter the translated values and descriptions for that locale.

For a table value set for which the underlying table supports multiple languages and for which the value set’s value column is based on a translated attribute of the underlying table, you can define translated values using the maintenance task for the underlying table. For more information on using multilanguage support features, see the Oracle Fusion Applications Developer’s Guide.

**FAQs for Manage Applications Core Value Sets**

**What happens if a value set is security enabled?**

Value set security is a feature that enables you to secure access to value set values based on the end user’s role in the system.

As an example, suppose you have a value set of US state names. When this value set is used to validate a flexfield segment, and users can select a value for the segment, you can use value set security to restrict them to selecting only a certain state or subset of states based on their assigned roles in the system.

For example, Western-region employees may choose only California, Nevada, Oregon, and so on as valid values. They cannot select non-Western-region states. Eastern-region employees may choose only New York, New Jersey, Virginia,
and so on as valid values, but cannot select non-Eastern-region states. Value set security is implemented using Oracle Fusion Applications data security.

**How can I set a default value for a flexfield segment?**

When you define or edit a flexfield segment, you specify a default value from the values provided by the value set assigned to that segment.

You can set the default value for a descriptive flexfield segment to be a parameter, which means the entity object attribute to which the chosen parameter is mapped provides the initial default value for the segment.

You can set the default value to be a constant, if appropriate to the data type of the value set assigned to the segment.

In addition to an initial default value, you can set a derivation value for updating the attribute’s value every time the parameter value changes. The parameter you choose identifies the entity object source attribute. Any changes in the value of the source attribute during runtime are reflected in the value of the segment.

If the display type of the segment is a check box, you can set whether the default value of the segment is checked or unchecked.

**Manage Applications Core Descriptive Flexfields**

**Flexfields: Overview**

A flexfield is an extensible set of placeholder fields in application pages that are associated with a business object. Each segment of the flexfield corresponds to a single application field, such as a segment of a key identifying a particular purchase, or the components of a student’s contact information, or the features of a product in inventory.

Using descriptive and extensible flexfields, you can extend business objects to capture data that wouldn’t otherwise be tracked by the application. If you need to add custom fields to a business object to meet your enterprise-specific requirements, configure the flexfield to have one segment for each needed field.

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

Flexfields let you meet enterprise requirements without changing the data model. Different data can be captured on the same database table. Each segment captures a single atomic value, has a name, and maps to a pre-reserved column in the application database.

You can use a flexfield to extend a business object if it has been registered for use on that object. Application developers create a flexfield and register it so that it is available for configuration. Administrators and implementation consultants set up or configure segments and other properties of the available flexfields. End users see flexfield segments as fields or attributes of information displayed in the application user interface. They enter a value for the attribute. The value may be
selected from a list of valid values or entered as free-form text that complies with formatting rules.

The following aspects provide an overview of flexfields:

- Accessing flexfields and flexfield management tasks
- Types of flexfields
- Flexfield segments
- Value sets
- Structure and context
- Deployment
- Run time appearance

**Accessing Flexfields and Flexfield Management Tasks**

You can view flexfields on a page where they occur using the Highlight Flexfields feature. You can access flexfield management tasks directly from a highlighted flexfield, through product-specific flexfield management tasks, or by starting in the Setup and Maintenance Overview page which is available from the Navigator or the Administration menu.

For lists of flexfields, see assets with the Flexfield: Descriptive, Flexfield: Extensible, or Flexfield: Key type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsoer.oracle.com).

**Types of Flexfields**

The following three types of flexfields are available in Oracle Fusion Applications and provide a means to customize applications features without programming.

- Key
- Descriptive
- Extensible

For example, in Oracle Fusion Financials, key flexfields represent objects such as accounting codes and asset categories. Generally, correct operations of a product depend on key flexfield setup. In Oracle Fusion Payables, a descriptive flexfield lets you collect custom invoice details fields on an invoices page. You can implement these fields, which are descriptive flexfield segments, as context-sensitive so they appear only when needed on a row-by-row basis when specific contextual information is met. Extensible flexfields are similar to descriptive flexfields, but provide additional advanced features. Generally, setup of descriptive and extensible flexfields is optional because their segments capture custom fields needed beyond the predefined fields.

**Segments**

Each field that you configure using flexfields is a flexfield segment. Segments represent attributes of information. They can appear globally wherever the flexfield is implemented, or based on a structure or context.

You define the appearance and meaning of individual segments when configuring a flexfield.
A key flexfield segment commonly describes a characteristic of the entity identified by the flexfield, such as a part number structured to include information about the type, color, and size of an item. A descriptive flexfield segment represents an attribute of information that describes a characteristic of the entity identified on the application page, such as details about a device containing components, some of which are globally present on the page while others are contextually dependent on the category of the device.

**Value Sets**

A value set is a named group of values that can be used to validate the content of a flexfield segment.

You configure a flexfield segment with a value set that establishes the valid values that an end user can enter for the segment. You define the values in a value set, including such characteristics as the length and format of the values. You can specify formatting rules, or specify values from an application table or predefined list. Multiple segments within a flexfield, or multiple flexfields, can share a single value set.

**Structure and Context**

Key flexfields have structure. Descriptive flexfields and extensible flexfields have context.

Each key flexfield structure is a specific configuration of segments. Adding or removing segments, or rearranging their order, produces a different structure. The database columns on which segments in different structures are based can be reused in as many structures as desired.

Descriptive flexfield segments can be context-sensitive, which means available to an application based on a context value rather than globally available wherever the flexfield appears. A descriptive flexfield context is a set of context-sensitive segments that store information related to the same context value. You define contexts as part of configuring a descriptive flexfield. End users see global segments, as well as any context-sensitive segments that apply to the selected context value.

Extensible flexfield segments are made available to an application based upon a category value. An extensible flexfield context serves as a container for related segments, used to organize the various segments that are applicable to a category value. You define contexts with context-sensitive segments and associate them to categories as part of configuring an extensible flexfield. End users see the segments displayed in subregions, one for each context associated to the selected category value.

In descriptive flexfields and extensible flexfields, the database columns on which context-sensitive segments are based can be reused in as many contexts as desired.

**Deployment**

A flexfield must be deployed to display its current definition in a run time application user interface. For example, if the deployment status is Edited, the flexfield segments may appear in the UI based on the flexfield definition at the time of last deployment, rather than the current definition.
Run time Appearance

In an application user interface, descriptive flexfield segments appear as label and field pairs or as a table of fields where the column headers correspond to the labels. The fields represent the flexfield segments and accept entered input or a selection from a list of choices that correspond to the segment’s assigned value set. Extensible flexfield segments appear grouped within labeled regions, where each grouping is a context and the region labels are the context names.

Use the Highlight Flexfields command in the Administration menu of the Setup and Maintenance work area to identify the location of the flexfields on the run time page. Flexfields in highlight mode display an Information icon button to access details about the flexfield, an Edit icon button to manage the flexfield, and an Add Segment icon button to add flexfield segments.

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

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

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.
4. 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**

The following Oracle Sales Cloud applications don’t support flexfields:

- Sales
- Marketing
- Customer Center
- Trading Community Architecture
- Order Capture

To add custom attributes to these applications, use Application Composer. For more information, see the "Editing an Object: Explained" section in 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.
Flexfields and Oracle Fusion Application Architecture: How They Work Together

Administrators configure flexfield segments to capture data that represents the values of attributes. Flexfield segments represent attributes of entities (business objects). Most business objects are enabled for descriptive flexfields. Some business objects are enabled for extensible flexfields.

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

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

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

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

- Integration
- Deployment
- Import and Export
- 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

- For Sales, Marketing, Customer Center, Trading Community Architecture, and Order Capture applications, use Application Composer to add custom attributes instead of using descriptive and extensible flexfields. For more information, refer to 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.

Manage Applications Core Descriptive Flexfields: Manage Flexfields

Managing Flexfields: Points to Consider

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

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

Consider the following aspects of managing flexfields:

- Registering flexfields
- Planning flexfields
- Configuring flexfields
- Enabling a flexfields segment for business intelligence
- Deploying flexfields
- Optionally changing a flexfield segment's appearance in a user interface page
- Identifying flexfields on a run time page and troubleshooting
Registering Flexfields

Application development registers flexfields so they are available to administrators and implementation consultants for configuration.

As part of registering a flexfield, application development reserves columns of entity tables for use in flexfields so an enterprise can capture segments to meet their business needs. Many flexfields are registered in Oracle Fusion Applications.

A flexfield must be registered before it can be configured.

For more information on registering flexfields, see Oracle Fusion Applications Developer’s Guide.

Planning Flexfields

Before you begin planning flexfields, determine what type is appropriate to your needs, and which business objects are available for customizing flexfields.

All flexfields consist of segments which represent attributes of an entity. The values an end user inputs for an attribute are stored in a column of the entity table.

Carefully plan flexfields before configuring them. Before configuring new segments for your flexfields, be sure to plan their implementation carefully.

If you have determined that a business object supports flexfields, and those flexfields have been registered, you can begin planning how to configure the flexfield for your needs. Note the code name of the flexfield you intend to configure so you can find it easily in the Define Flexfield activity.

In some cases you can customize how the flexfield appears on the page.

See Oracle Fusion Applications Help for specific products to determine any restrictions on using product-specific flexfields.

Configuring Flexfields

Administrators or implementers configure flexfields so they meet the needs of the enterprise. Some flexfields require configuration to make an application operate correctly.

You can configure flexfields using the following methods:

- Go to the manage flexfield tasks in the Setup and Maintenance work area.
- Use the Highlight Flexfields command in the Administration menu while viewing a run time page.
- Use the Configure Flexfield icon button to manage a flexfield, such as change a segment’s sequence number, or configure a flexfield segment’s business intelligence label.
- Use the Add Segment icon button to add descriptive flexfield segments and context values, or extensible flexfield segments.
Configuring a flexfield includes the following:

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

**Tip**

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

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

You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order based on the segments' sequence numbers. You cannot enter a number for one segment that is already in use for a different segment.

**Tip**

Consider numbering the segments in multiples, such as 4, 5, or 10, to make it easy to insert new attributes.

A flexfield column is assigned to a new segment automatically, but you can change the assignment before saving the segment. If you need to set a specific column assignment for a segment, create that segment first to ensure that the intended column isn’t automatically assigned to a different segment.

### Enabling a Flexfield Segment for Business Intelligence

You can enable flexfield segments for business intelligence if the flexfield is registered in the database as an Oracle Business Intelligence-enabled flexfield. For more information on enabling segments for business intelligence, see points to consider when enabling key and descriptive flexfield segments for business intelligence.

For extensible flexfield segments, you can’t assign labels and use equalization to prevent duplication.

### Deploying Flexfields

Once you have configured a flexfield, you must deploy it to make the latest definition available to run time users.
In the Define Flexfields tasks, you can deploy a flexfield using either of the following commands:

- The Deploy Flexfield command to deploy a flexfield to mainline. This is for general use in a test or production environment.
- The Deploy to Sandbox command to deploy a flexfield to sandbox. This is to confirm that the flexfield is correctly configured before deploying it to the mainline.

When using the **Add Segment** and **Edit Segment** tools for descriptive flexfields in Highlight Flexfields mode, you can use the Save and Deploy command to save your changes and deploy the flexfield to mainline.

Once deployed, the deployment status indicates the state of the currently configured flexfield relative to the last deployed definition.

### Optionally Changing a Flexfield Segment Appearance

The flexfield attributes that you define integrate with the user interface pages where users access the attributes' business object. Application development determines the UI pages where business objects appear and the display patterns used by default to render flexfield segments.

After a flexfield has been deployed to a mainline metadata services (MDS) repository so that it appears on application pages, you can customize it on a per-page basis using Page Composer. For example, you can hide a segment, change its prompt or other properties, or reorder the custom global attributes so that they are interspersed with the core attributes in the same parent layout.

You can only customize the appearance of descriptive and extensible flexfield segments in the UI page using Page Composer once the flexfield is deployed to the mainline.

If the Oracle Fusion applications are running in different locales, you can provide different translations for translatable text, such as prompts and descriptions. Enter translations by signing in using the locale that requires the translated text. You do this by selecting **Settings and Actions - Personalization - Set Preferences** in the global area and changing the text to the translated text for that locale.

### Identifying Flexfields on a Run time Page and Troubleshooting

The **Highlight Flexfields** command in the Administration menu of the Setup and Maintenance work area identifies the location of flexfields on the run time page by displaying an **Information** icon button for accessing details about each flexfield.

Even if a descriptive or extensible flexfield hasn't yet been deployed and no segments appear on the run time page in normal view, the flexfield appears in the Highlight Flexfield view for that page. In the case of descriptive flexfields, the segments as of the last deployment appear. **Highlight Flexfields** accesses the current flexfield metadata definition.

Use the highlighted flexfield's **Edit** icon button to manage flexfields directly. Alternatively, note a highlighted flexfield's name to search for it in the tasks for managing flexfields.
To examine a flexfield's configuration, export the deployed artifacts using the `exportMetadata WLST`.

For more information on creating flexfields and adding them to a UI page, see the Oracle Fusion Applications Developer's Guide.

For more information about customizing flexfield segment appearance with Oracle Composer, see guidance on customizing existing pages in the Oracle Fusion Applications Extensibility Guide.

**Flexfield Segment Properties: Explained**

Independent of the value set assigned to a segment, segments may have properties that affect how they are displayed and how they behave.

The following aspects are important in understanding:

- Display properties
- Properties related to segment values
- Properties related to search
- Range validation segments
- Rule validation of segment values
- Naming conventions

**Display Properties**

The following table summarizes display properties.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Whether the segment can be used.</td>
</tr>
<tr>
<td>Sequence</td>
<td>The order the segment appears in relation to the other configured segments.</td>
</tr>
<tr>
<td>Prompt</td>
<td>The string to be used for the segment's label in the user interface.</td>
</tr>
<tr>
<td>Display type</td>
<td>The type of field in which to display the segment.</td>
</tr>
<tr>
<td>Checked and unchecked values</td>
<td>If the display type is check box, the actual values to save. For example, Y and N or 0 and 1.</td>
</tr>
<tr>
<td>Display size</td>
<td>The character width of the field.</td>
</tr>
<tr>
<td>Display height</td>
<td>The height of the field as measured in visible number of lines when the display type is a text area.</td>
</tr>
<tr>
<td>Read only</td>
<td>Whether the field should display as read-only, not editable text.</td>
</tr>
<tr>
<td>Description help text</td>
<td>The field-level description help text to display for the field. Use description help text to display a field-level description that expands on or clarifies the prompt provided for the field.</td>
</tr>
<tr>
<td></td>
<td>If description help text is specified, a Help icon button is displayed next to the field in the run time application. The description help text is displayed when the user hovers over the Help icon button.</td>
</tr>
</tbody>
</table>
Properties Related to Search

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

Range Validation of Segments

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

The conditions for range validation are as follows:

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

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

Rule Validation of Segment Values

Validation rules on descriptive and extensible flexfield segments determine how an attribute is validated. The value entered for an attribute on a business object may need to match a specified format or be restricted to a list of values. Use a value set to specify the validation rules.
Value set validation is required for global segments and context-sensitive segments, and optional for context segments. In the case of context segments, the application may validate an input value instead of the value set validating the input value against the context segment. However the application input values must match exactly the valid context segment values. If the context segment values are a superset or subset of the input values, you must assign a table-validated value set or independent value set to validate context values.

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

Naming Conventions

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

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

Flexfields and Value Sets: How They Work Together

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

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

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

- Defining value sets
- Shared value sets
- Deployment

Defining Value Sets

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

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>Parameter</td>
<td>Yes</td>
<td>No</td>
<td>The default segment value is the parameter's default value</td>
<td>N/A</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and same as default value</td>
<td>The default segment value is the parameter's default and derivation value</td>
<td>The changed parameter derivation value updates segment value</td>
</tr>
<tr>
<td>Parameter</td>
<td>Yes</td>
<td>Yes, and different from default value</td>
<td>The default segment value is the parameter's default value</td>
<td>The changed parameter default value doesn't update segment value. Only the changed derivation value updates the segment value.</td>
</tr>
</tbody>
</table>

**Flexfield Usages: Explained**

Usage affects various aspects of flexfields. The usage of the flexfield is set when the flexfield is registered and specifies the application and table with which the flexfield is associated.

Entity usage indicates the table containing the segments of a flexfield.

A flexfield can have multiple usages. The first table registered for a flexfield is the master usage. Segments are based on the master usage, and other usages of the same table for the same flexfield use the same segment setup, though the column names optionally may have a differentiating prefix.

**Extensible Flexfields**

You can configure different behavior for extensible flexfield contexts at the usage level. The usage of an extensible flexfield context determines in which scenarios or user interfaces the segments of a context appear to end users. For example, if a Supplier page displays an extensible flexfield’s supplier usage and a buyer page displays that same extensible flexfield’s buyer usage, a context that is associated...
to the supplier usage but not the buyer usage displays only on the supplier page and not the buyer page.

**Value Sets**

The usage of value sets specifies the flexfields having segments where the value set is assigned.

**FAQs for Manage Applications Core Descriptive Flexfields: Manage Flexfields**

**Why did my flexfield changes not appear in the run time UI?**

The ADF business components or artifacts of a flexfield, which are generated into an metadata services (MDS) repository when the flexfield is deployed, are cached within a user session. You must sign out and sign back in again to view flexfield definition changes reflected in the run time application user interface page.

A flexfield’s deployment status indicates whether the flexfield segments as currently defined in the metadata are available to end users. The flexfield segments seen by end users in the run time correspond to the flexfield definition that was last deployed successfully.

**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.

**Manage Applications Core Descriptive Flexfields: Deploy Flexfields**

**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>deployFlexForApp</td>
<td>Deploys all flexfields for the specified enterprise application. Only flexfields whose status is other than deployed are affected by this command unless the option is enabled to force all flexfields to be deployed regardless of deployment status. Initial application provisioning runs this command to prime the MDS repository with flexfield artifacts.</td>
</tr>
<tr>
<td>deployFlex</td>
<td>Deploy a single flexfield regardless of deployment status</td>
</tr>
<tr>
<td>deployPatchedFlex</td>
<td>Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>deleteFlexPatchingLabels</td>
<td>Displays MDS label of flexfield changes for viewing and deleting patching labels.</td>
</tr>
<tr>
<td>ValidateFlexDeploymentStatus</td>
<td>Displays list containing flexfields that aren't deployed or failed deployment.</td>
</tr>
</tbody>
</table>

Executing these commands outputs a report at the command line. The report provides the following information for every flexfield that is processed.

- Application identity (APPID)
- Flexfield code
- Deployment result, such as success or error

In case of errors, the report lists the usages for which the errors were encountered. If a run time exception occurs, the output displays the traceback information. For each WLST flexfield command, adding the `reportFormat='xml'` argument returns the report as an XML string.

Consider the following aspects of command line deployment.

- Preparing to use the WLST flexfield commands
- Using the `deployFlexForApp` command
- Using the `deployFlex` command
- Using the `deployPatchedFlex` command
- Using the `deleteFlexPatchingLabels` command
- Using the `validateFlexDeploymentStatus` command
- Exiting the WLST and checking the results

**Preparing To Use the WLST Flexfield Commands**

You can only execute the WLST flexfield commands on a WebLogic Administration Server for a domain that has a running instance of the Oracle Fusion Middleware Extensions for Applications (Applications Core) Setup application.

For more information on deploying the Applications Core Setup application, see the Oracle Fusion Applications Developer's Guide.

Ensure that the AppMasterDB data source is registered as a JDBC data source with the WebLogic Administration Server and points to the same database as the ApplicationDB data source.

Start the WebLogic Server Tool (WLST) if it isn’t currently running.

UNIX:

```
sh $JDEV_HOME/oracle_common/common/bin/wlst.sh
```

Windows:

```
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.

```
deployFlex('flex_code', 'flex_type', ['force_Complete_EFF_Deployment'])
```

Where, forceCompleteEFFDeployment=None

Using the deployPatchedFlex Command

Use the `deployPatchedFlex` command for situations where the patching framework doesn’t invoke the command, such as when an application has been patched offline.

If the installation is multi-tenant enabled, the command deploys all patched flexfields for all enterprises. This command isn’t intended to be invoked manually.

Check with your provisioning or patching team, or the task flows for managing flexfields, to verify that the flexfield has a Patched deployment status.

From the WLST tool, execute the following command to deploy the artifacts to the MDS partition.

```
deployPatchedFlex()
```

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

```
deployPatchedFlex(mode='RETRY')
```

Using the deleteFlexPatchingLabels Command

Whenever you deploy flexfield changes to MDS using the `deployPatchedFlex()` WLST command, an MDS label is created in the format FlexPatchingWatermarkdate+time. Use the `deleteFlexPatchingLabels` command to inquire about and delete these labels.

From the WLST tool, execute the `deleteFlexPatchingLabels()` command with no arguments to delete the flexfield patching labels.

To output a list of flexfield patching labels, execute the command with the `infoOnly` argument, as follows:

```
deleteFlexPatchingLabels(infoOnly='true')
```
Using the validateFlexDeploymentStatus Command

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

validateFlexDeploymentStatus()

Use this command to verify that all flexfields in the current instance of provisioned Java EE applications are deployed.

Exiting the WLST and Checking the Results

To exit the tool, execute the following command.

disconnect()

Optionally, sign into the application, access user interface pages that contain flexfields, and confirm the presence of flexfields for which configuration exists, such as value sets, segments, context, or structures.

Manage Applications Core Descriptive Flexfields: Manage Descriptive Flexfields

Descriptive Flexfields: Explained

Descriptive flexfields provide a way to add custom attributes to entities, and define validation and display properties for them. These attributes are generally standalone. They don’t necessarily have anything to do with each other and 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 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>
<tr>
<td>List of Values</td>
<td>Independent</td>
</tr>
<tr>
<td>Radio Button Group</td>
<td>Independent</td>
</tr>
<tr>
<td>Text Field With Search</td>
<td>Independent</td>
</tr>
<tr>
<td>Text box</td>
<td>Format Only</td>
</tr>
<tr>
<td>Text area</td>
<td>Format Only</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Format Only</td>
</tr>
</tbody>
</table>

Tip

After you add a context value, refresh the page to see the new value.

Usages

Descriptive flexfield usages allow for the same definition to be applied to multiple entities or application tables, such as a USER table and a USER_HISTORY table. Descriptive flexfield tables define the placeholder entity where the flexfield segment values are stored once you have configured the descriptive flexfield. When you configure a flexfield, the configuration applies to all its usages.

Parameters

Some descriptive flexfields provide parameters, which are attributes of the same or related entity objects. Parameters are public arguments to a descriptive flexfield. Parameters provide outside values in descriptive flexfield validation. You use parameters to set the initial value or derivation value of an attribute from external reference data, such as a column value or a session variable, rather than from user input. Parameters can be referenced by the logic that derives the default segment value, and by table-validated value set WHERE clauses.

Delimiters

A segment delimiter or separator visually separates segment values when the flexfield is displayed as a string of concatenated segments.
Initial Values

The SQL statement defining an initial value must be a valid statement that returns only one row and a value of the correct type.

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use the following bind variables in the WHERE clause of the SQL statement.
  
  - `: (SEGMEN'T. <segment_code>)` : Identifies a segment in the same context.
  
  - `: (CONTEXT. <context_code>; SEGMENT. <segment_code>)` : Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it cannot be a multiple-row context.
  
  - `: (VALUESET. <value_set_code>)` : Identifies the closest prior segment in the same context that is assigned to the specified value set.
  
  - `: (FLEXFIELD. <internal_code>)` : Identifies a flexfield.

For more information about using bind variables, see the help for value sets.

Business Intelligence

Selecting a global, context, or context-sensitive segment’s BI Enabled checkbox specifies that the segment is available for use in Oracle Business Intelligence.

When the flexfield is imported into Oracle Business Intelligence, the label you selected from the BI Label dropdown list equalizes the segment with segments in other contexts, and maps the segment to the logical object represented by the label.

Enabling Descriptive Flexfield Segments for Business Intelligence: Points to Consider

A descriptive flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segments. When a global, context, or context-sensitive segment is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled flexfield segments:

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component
• Mapping attributes of flattened business components to logical objects in Oracle BI
• Managing the labels that map segments to logical objects in Oracle BI

After you deploy a business intelligence-enabled flexfield, use the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process to import the flexfield changes into the Oracle Business Intelligence repository. Users can make use of the newly-generated attributes in business intelligence applications. For example, a user can generate a report that includes attributes added by the descriptive flexfield. For additional information about logical objects and import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

Flattening

When you deploy a business intelligence-enabled descriptive flexfield, the deployment process generates an additional set of flattened Application Development Framework (ADF) business components in addition to the usual ADF business components and ADF faces run time artifacts that are generated during deployment. The flattened business components include attributes for business intelligence-enabled segments only. Flattening means each custom column in each context shows up as an attribute in an Oracle Business Intelligence folder.

Flattened components include one attribute for the BI-enabled context-segment, and one attribute for each business intelligence-enabled global segment. For BI-enabled context-sensitive segments, consider the following:

• If you assigned a label to the segment, the flattened components include an additional single attribute representing segments with that label.

• If you didn’t assign a label, the flattened components include a discrete attribute for each BI-enabled context-sensitive segment in each context.

Mapping to Logical Objects in Business Intelligence

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence.

If you assign a label to any set of context-sensitive segments that serve the same purpose in different contexts, you can consolidate or equalize the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, a United States context might have a Passport segment and a Canada context might have Visa segment. If you assign the NationalID segment label to both the Passport and Visa segments, they are equalized into the same NationalID attribute in the flattened business component.

Non-labeled context-sensitive segments aren’t equalized across context values, so the flattened components include a separate attribute for each context-sensitive segment for each context value.

Note

It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.
Assign a label to a global segment, context segment, or context-sensitive segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

**Note**

Assigning a label to a context-sensitive segment serves to equalize the attribute across contexts, as well as map the equalized attribute to business intelligence.

**Managing Labels**

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across contexts.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn’t exist in business intelligence, you must manually map the segment to the desired logical object when importing into Oracle Business Intelligence.

In addition, context-sensitive segments without labels cannot be equalized across context values. The flattened components include a separate attribute for each non-labeled context-sensitive segment in each context.

**Importing to Oracle Business Intelligence Repository**

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION_APPS_BI_APPID user.

**Note**

To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator’s Guide.

**Tip**

When you import a flexfield into the Oracle Business Intelligence repository, you see both `<name>` and `<name>_c` attributes for each segment, along with some other optional attributes. The `<name>` attribute contains the value. The `<name>_c` attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.
FAQs for Manage Applications Core Descriptive Flexfields

Can I display the context segment in the project Cost Collection flexfield?

No. The context segment is predetermined for each page. Displaying it and changing the value may result in capture of data that is not applicable for the current transaction.

Manage Applications Core Attachment Entities

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</td>
</tr>
<tr>
<td></td>
<td>(FND_READ_APPLICATION_ATTACHMENT_DATA)</td>
</tr>
<tr>
<td>Add or Update</td>
<td>Update Application Attachment</td>
</tr>
<tr>
<td></td>
<td>(FND_UPDATE_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 Applications Core Common Reference Objects: Manage Applications Core Attachment Categories**

**What's an attachment category?**

An attachment category is used to classify and secure attachments. Each attachment user interface must be defined with at least one category for users.
to be able to add attachments. If there are multiple categories, users can view
them and select one when adding attachments. For example, attachments for an
expense report can be categorized as receipts, scanned invoice images, and so on.

You can also associate roles with categories to determine user access and actions
for attachments, based on the categories assigned to the attachment entity. For
example, security for expense report attachments can be based in part on the
categories assigned to the expense report attachment entity. You can define
multiple categories per module, and add and manage custom categories for your
own purposes. For more information on attachment category data security, see
the Oracle Fusion Applications Developer's Guide.

Use the Manage Attachment Categories page, which you can access by starting
in the Setup and Maintenance Overview page and searching for the Manage
Attachment Categories task.
Define Common Project Execution Options

Manage Project User Provisioning

Provisioning Access to Project Execution Management Applications: Overview

Provisioning access to Project Execution Management applications is the process of registering a user account in Oracle Identity Management for a resource and assigning project-related enterprise roles to the resource. A resource’s access to work areas in Project Execution Management applications depends on the roles that you assign to the resource.

Use the Manage Project User Provisioning page to request user accounts and assign enterprise roles for project enterprise labor resources to enable the resources to sign into Project Execution Management applications to plan projects, manage resources, review, track, and collaborate on work. During implementation you can provision a set of users and assign the Project Application Administrator role so that these administrators can initiate the provisioning process for the rest of the project enterprise labor resources.

Resources to Provision

A resource that you provision typically falls into one of these categories:

- Resource is an HCM person in Oracle Fusion HCM and is a project enterprise labor resource in Oracle Fusion Project Management.

User accounts for resources in this category are typically created in Oracle Fusion HCM, and the HCM person is associated with a project enterprise labor resource when you create the resource in Oracle Fusion Project Management. Assign the appropriate project-related roles to these resources so they can perform project management, project resource management, project collaborative planning, and progress functions in Project Execution Management applications.

Important
You cannot create a user account on the Manage Project User Provisioning page for an existing HCM person. HCM persons are registered in Oracle Fusion HCM.

- Resource is a project enterprise labor resource in Oracle Fusion Project Management, but is not an HCM person.

  You can maintain resource details and add resources to projects in Oracle Fusion Project Management even if the resources are not employees or contingent workers in Oracle Fusion HCM. However, if you want these resources to access Project Execution Management applications to administer or manage projects or resources, or track and report on work, then you must create user accounts to register the resources in Oracle Identity Management, and assign project-related enterprise roles to the resources. Select the existing project enterprise labor resource on the Manage Project User Provisioning page and initiate the provisioning process.

- Resource is an HCM person in Oracle Fusion HCM, but is not a project enterprise labor resource in Oracle Fusion Project Management.

  You can assign project-related enterprise roles to resources who have user accounts that were created in Oracle Fusion HCM. However, these resources do not appear as project enterprise labor resources in Project Execution Management applications until you create the resources in Oracle Fusion Project Management.

**Enterprise Roles**

You can provision the following predefined enterprise roles to resources:

- **Project Application Administrator**: Collaborates with project application users to maintain consistent project application configuration, rules, and access.

- **Project Execution**: Manages projects in Project Execution Management applications.

**Note**

The Project Manager job role does not include the Project Execution role by default.

- Resource Manager: Manages a group of resources who are employees or contingent workers. Monitors the utilization of resources and manages the assignment of resources to work on projects. Collaborates with project managers to find suitable resources to fulfill project resources requests.

- Team Collaborator: Performs, tracks, and reports progress on project and nonproject work. Collaborates with other team members or project managers to perform project tasks and to-do tasks.

In addition, you can provision custom job roles for resources. For example, you can provision a Custom Team Member role that contains a different set of security permissions than the Project Team Member role.
Manage Project User Provisioning

On the Manage Project User Provisioning page:

- When you select the **Request user account** check box, the roles that you specified to provision by default appear in the Role Details table for the resource.

- Select the **Assign administrator role** check box to assign the Project Application Administrator role to the resource. This action adds the Project Application Administrator role to the Role Details table.

- Add the Resource Manager role to the Role Details table if the resource will perform functions in Oracle Fusion Project Resource Management.

- You can add or remove roles for a resource with an existing user account. Use the Edit feature to add roles. Use the Actions menu to remove roles.

**Note**

You must wait until any previous provisioning request is complete for this resource before you add or remove roles for the resource.

- Use the Assign Resource as Project Manager action in the Search Results region to add a resource to a project as a project manager. If the resource is an existing project user, does not have a pending provisioning request, and is not assigned the Project Execution role, then the application provisions the Project Execution role for the resource.

- Click the link in the **Last Request Status** column to view the details of the most recent provisioning action for a resource.

Manage Project Enterprise Resources

You can request a user account when you create or edit a project enterprise labor resource who is not associated with an HCM person. This action provisions the default role assignments for the resource and sends the resource an e-mail notification.

**User Account and Provisioning Statuses**

A user account status indicates whether a resource can access the application. The statuses are:

- **Active**: User is active and can access the application
- **Inactive**: User is inactive and cannot access the application

A user provisioning status indicates the state of the user provisioning process. The statuses are:

- **Requested**: Role provisioning is requested for a resource
- **Completed**: Process completed without errors or warnings
- **Failed**: Process failed because of errors or warnings
- **Partially completed**: Process is partially completed
- **Pending**: Process is in progress
Assigning a Resource as a Project Manager from the Manage Project User Provisioning Page: Explained

A project application administrator can use the Assign Resource as Project Manager action in the Manage Project User Provisioning page - Search Results region to add a resource to a project as a project manager. For example, assume that a project team member will take over the project manager responsibilities while the manager is away on leave. The administrator can add the team member as a project manager and provision the Project Execution role for the team member.

When the administrator adds the resource as a project manager:

- If the resource is an existing project user, does not have a pending provisioning request, and is not assigned the Project Execution role, then role provisioning is requested to assign the Project Execution role to the resource. The resource is added to the project as a project manager with a Planning Only assignment status.
- If the resource is an existing project user with the Project Execution role, then the resource is added to the project as a project manager and no provisioning occurs.
- If a resource is not an existing project user, or if a resource has a pending provisioning request, then you cannot assign the resource as a project manager from the Manage Project User Provisioning page.

Manage Project Management Roles

Project Roles in Project Execution Management Applications: Explained

A project role is a classification of the relationship that a person has to a project, such as project manager, functional consultant, or technical lead. A project role defines the type of work that a person performs on a project, and allows access to project management information for the project manager role.

Project manager and project team member are predefined project roles that you cannot edit or delete. You can create additional project roles to meet the needs of your organization. However, you cannot delete a project role that is designated as a resource’s primary project role, specified on a project resource request, or assigned to a resource on a project.

Project roles are used for the following purposes:
• To identify the type of work that a person performs on project assignments
• To set up default resource qualifications
• As criteria when searching for resources to fulfill project resource requests
• As a resource’s primary project role

Project Assignments
When you add a resource to a project, you select a project role to identify the type of work that the resource will perform on the project. The default project role is Project Team Member for resources or resource placeholders that you add directly to the Manage Project Resources page in the Project Management work area.

When you fulfill a project resource request in the Project Resources work area and create an assignment for the resource, the project role specified on the request is the default project role on the assignment. You can change the project role on the Assign Resource page before you submit the assignment for approval.

Default Resource Qualifications
On the Manage Project Roles page, select a set of default qualifications, proficiencies, and keywords for each project role. Default qualifications, proficiencies, and keywords that you associate with a project role automatically appear as requirements on a project resource request when you select the project role for the request.

Project Resource Requests
When searching for resources to fulfill a project resource request on the Search and Evaluate Resources page, you can filter the resource search results by project role to focus the results. Resources that have the selected project role filter as their primary project role will appear in the results. The number next to the filter indicates the number of resources in the results that have the primary project role.

Primary Project Roles
You can designate a primary project role for a resource that represents the work that the resource typically performs on project assignments.

You can use the resource’s primary project role in the following areas in Oracle Fusion Project Resource Management:
  • As a resource search option filter when viewing resources on the Search and Evaluate Resources page
  • When viewing resource information on the Resource Details page
  • When comparing the attributes of multiple resources against the requirements specified in the project resource request on the Compare Resources page
  • As an attribute value to assign to new resources that the Maintain Project Enterprise Labor Resources process creates
  • As search criteria when searching for a project enterprise labor resource to designate as a resource pool owner on the Manage Resource Pools page
• As advanced search criteria when searching for resource pool members on the Manage Resource Pools page
• As column to sort open project resource requests on the Resource Manager Dashboard

Define Project and Resource Calendars

Project and Resource Calendars: Explained

Project and resource calendars define the standard working and nonworking time for projects and resources. Working times are the days and total hours in a calendar during which work can occur. Nonworking times in a calendar are days and total hours for which work should not be scheduled, such as weekends and holidays.

Features of project and resource calendars include:

• Default calendars
• Project calendars
• Resource calendars
• Standard calendar

Default Calendars

Your implementation team selects a default project calendar and resource calendar on the Define Project Management Implementation Options page. The default project calendar is automatically assigned to new projects, and the default resource calendar is automatically assigned to new project enterprise labor resources.

If the default project calendar does not meet the schedule requirements for a specific project, a project manager can select another calendar during project creation. Similarly, if the default resource calendar does not meet the scheduling requirements for a specific project enterprise labor resource, the project application administrator can assign another calendar to the resource.

Project Calendars

A project manager can select the calendar for a project during project creation or when managing the project plan.

The working days and total hours in the calendar are used to schedule the project plan. The calendar also works with the project start and finish dates to determine the project duration.

Resource Calendars

The project application administrator can select a resource calendar when creating or editing a project enterprise resource. The working days and total working hours in a resource calendar are used to determine resource availability.

Oracle Fusion Project Resource Management uses the resource calendar to determine a resource’s available hours for a project. The Update Resource
Utilization Data process calculates the resource's projected utilization with the available hours.

**Standard Calendar**

Oracle Fusion Project Execution Management provides a standard calendar of 8 hours per day, 5 days per week that you can assign to projects and resources.

**Schedule Components: How They Fit Together**

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

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

**Shift**

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

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

Exception

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

Schedule

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

Managing Shifts: Examples

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

Managing Time Shifts

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

Managing Time Shifts with Punch Details

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

Managing Time Shifts with Flexible Details

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

**Managing Duration Shifts**

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

**Managing Elapsed Shifts**

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

**Managing Workday Patterns: Examples**

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

**Managing Time Workday Patterns**

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

**Managing Duration Workday Patterns**

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

**Managing Elapsed Workday Patterns**

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

FAQs for Define Project and Resource Calendars

What happens if project resource request dates are outside the resource calendar effective dates?

The application cannot calculate a resource’s available capacity if the project resource request dates are outside of the resource calendar effective dates. The resource will have no Available Capacity score.

Manage Project Enterprise Resources

Project Enterprise Labor Resource Components: How They Work Together

A project enterprise labor resource is a resource that you can assign to multiple projects. If you manage resource availability and staffing in Oracle Fusion Project Resource Management, use project enterprise labor resources to fulfill project resource requests.

Consider the following attributes when creating a project enterprise labor resource.

- Select the type attribute
- Select the resource calendar
- Select the primary project role
- Associate an employee or contingent worker
- Manage resource availability and staffing
- Select a resource pool
- Manage rates and amounts

Selecting the Type Attribute

You can add both labor and expense type resources to a project. Project enterprise labor resources are the Labor resource type.

Selecting the Resource Calendar

Resource calendars are used to determine resource availability, schedule tasks, and assign resources to tasks.

The main purpose of the calendar is for daily scheduling of work using hours per day as the unit. The duration of tasks is measured in days and not hours.

A predefined calendar with eight hours per day and five days per week with no holidays is used as a default if no calendar is available for a project or resource.
Selecting the Primary Project Role

Available values are based on Oracle Fusion Project Management project roles. The primary project role represents the role that a resource most often fulfills on a project.

Note

The default value is Project Team Member.

Associating an Employee or Contingent Worker

An HCM person is a named person in the HCM database with a person type of Employee or Contingent Worker and an active assignment. In Oracle Fusion Project Resource Management, if you associate an HCM person with a project enterprise labor resource and manage the resource’s availability and staffing, then the resource is eligible to fulfill project resource requests.

Managing Resource Availability and Staffing

Select this option for the resource to be eligible to fulfill project resource requests. If you deselect this option after selecting it, the resource availability and staffing is no longer managed in Oracle Fusion Project Resource Management, and the following changes occur:

- If the resource is not used in Project Resource Management, then the resource is no longer eligible to fulfill project resource requests and is removed completely from pools.
- If the resource is used in Project Resource Management, then resource is no longer eligible to fulfill project resource requests and remains in pools for historical reporting purposes. If the resource is already associated with an unfulfilled request, then the resource is no longer eligible to be assigned to fulfill the request.
- The application sends the resource pool owner a notification that the resource is no longer eligible to fulfill project resource requests. This notification is for information only.

Selecting a Resource Pool

You must select a resource pool if you manage the resource availability and staffing in Project Resource Management.

Note

The default value is the Resources with No Pool Membership pool.

Managing Rates and Amounts

The rate and amount types on a project are:

- Bill Rate: The rate for a unit of work that determines the invoice or revenue recognized amount for a resource on a project.
- Cost Rate: The rate for a unit of work that determines the cost for a resource on a project.
• Bill Amount: The bill amount for a resource allocated to a project, calculated by multiplying the labor effort by the project resource bill rate.

• Cost Amount: The cost amount for a resource allocated to a project, calculated by multiplying the labor effort by the project resource cost rate.

You can assign cost and bill rates to a resource on the project enterprise labor resource definition. When you add a resource to a project, the application copies the resource cost and bill rates to the project if the rate currency is the same as the project currency.

You can edit planning-only resource rates directly on the project, and the edits do not affect the rates on the resource definition. You cannot edit the rates on the project for resources in an Assigned status.

Import Project Enterprise Resources

The Import Project Enterprise Resources process imports the resources from third-party applications in the interface table to Oracle Fusion Project Management. The process creates project enterprise resources for example, labor or expense type, based on the data loaded into the PJT_PRJ_ENT_RES_INTERFACE open interface table.

Note

You can load data to interface tables using predefined templates and the Load Interface File for Import scheduled process, which are both part of the External Data Integration Services for Oracle Cloud feature. For more information, see the Documentation tab for the Load Interface File for Import process in Oracle Enterprise Repository for Oracle Fusion Applications.

From the navigator, select the Scheduled Processes menu and specify the process name, date and time to run the Import Project Enterprise Resources process.

The process validates the resources and creates project enterprise resources in Oracle Fusion Project Management. The exceptions for the invalid resources are tracked in a report for the Import Project Enterprise Resources process. Review the errors and fix the issues. Load the data from CSV file into the interface table again and resubmit the process.

Import Project Enterprise Resources Execution Report

After the Import Project Enterprise Resources process completes, review the output report to view the successful transactions, errors, and warnings. View the summary report and detailed report and track the details by resource type, labor and expense type. If errors exist, review and resolve the issues before submitting the process again.

Project Enterprise Expense Resources: Explained

Use project enterprise expense resources to manage expense type resources that can be assigned to projects or tasks in the Project Execution Management application in Oracle Fusion Project Portfolio Management. You can use
Define Common Project Execution Options

predefined project enterprise resource expense type options, create your own expense resource type, or create from expenditure type resources existing in project financial management in Oracle Fusion Project Foundation.

Consider the following attributes when creating a project enterprise expense resource.

- Select the type attribute
- Create from expenditure type option
- Select the expenditure type
- Create an expenditure type resource
- Manage unit of measure

Selecting the Type Attribute

You can add an expense type resource to a project. Project enterprise expenditure type resources are the Expense resource type.

Creating from Expenditure Type Option

The Create from expenditure type option is used to create expense type resources from expenditure type planning resources created in Oracle Fusion Project Foundation.

Selecting the Expenditure Type

Expenditure types can be used to track various expenses on projects or tasks. You can link to existing expenditure types for expenses that are used regularly including hotel, airfare, and meals.

You cannot edit predefined expenditure types.

Creating an Expenditure Type Resource

Identify unique expense requirements for your organization. For example, on a project you may want to track cell phone expenses separately from home phone expenses.

Managing Unit of Measure

You cannot edit the unit of measure for expense resources on a project. The Default value is Currency. For example, if you have a project related expense for a car rental in San Francisco, the currency would be 35 dollars based on the project currency being USD. Hotel expenses in Bangalore on a project with currency as Rupees could be INR; 5000.

FAQs for Manage Project Enterprise Resources

How can I remove the Project Application Administrator role from a resource?

Use the Actions menu in the Manage Project User Provisioning page - Edit Project User window - Role Details table to revoke a project-related role from a project user.
Can I edit the name and e-mail for a project enterprise labor resource?

You can edit the name and e-mail for a resource who was created in Oracle Fusion Project Management, does not have a user account, and is not associated with an HCM person.

You can no longer edit the name and e-mail once you start the process to create a user account and provision roles for the resource.

What's the difference between a project enterprise labor resource and a managed project enterprise labor resource?

A project enterprise labor resource is a resource that you can assign to multiple projects.

A managed project enterprise labor resource is a project enterprise resource who is associated with a named person in HCM, and whose availability and staffing are managed in Oracle Fusion Project Resource Management. You use managed project enterprise labor resources to fulfill project resource requests.
FAQs for Manage Project Management Implementation Options

What's the difference between project calendars and resource calendars?

Project calendars determine standard working and nonworking times for resources working on projects. For example, a project calendar can designate the total number of working hours per day and any holidays that occur during the project dates. A project calendar is used to schedule project tasks, and to estimate project duration.

Resource calendars determine the total available working hours for resources.

Note

Your implementation team defines the default project and resource calendars that are automatically assigned to new projects and resources. However, a project manager can assign another calendar to a project. Similarly, a project application administrator can assign a different calendar to a resource.

Can I change the default value for the resource calendar that appears when I create a project enterprise labor resource?

Yes. On the Define Project Management Implementation Options page, you select a default resource calendar that the application automatically assigns to new project enterprise labor resources. If the default resource calendar does not meet the requirements for a specific project enterprise labor resource, then you can assign a different calendar to the resource.
Why do some tasks have exceptions for progress entries?

A team member entered a proposed start date, proposed finish date, or an increase to the planned hours for the task that exceeds the progress thresholds defined in the project management implementation options.

Manage Enterprise Project and Task Codes

Enterprise Project Codes and Task Codes: Explained

Project codes and task codes enable you to capture information at the project and task level that is specific to your organization. Create project and task codes based on one of the following data types to determine how project managers and project team members enter information.

- Value sets
- Free form text
- Numeric

Value sets

Define up to 40 project or task codes based on value sets. When you define a value set, enter at least one value for project managers and team members to select when they create projects or tasks. You can specify a sequence for the values to appear in a specific order.

Free form text

Define up to 20 text fields to enable users to enter alphanumeric information for project or task codes.

Numeric

Define up to 10 project or task codes that allow entry of numeric information. You cannot assign the same code to both project and task codes. After you use a code at the task level, you cannot use it at the project level.

Manage Custom Attributes for Issues and Deliverables

Customizing Project Execution Management Applications: Explained

Customize view objects for Project Execution Management applications in Oracle Fusion Project Portfolio Management using Data Composer and Page Composer. The following sections describe the aspects of customization:

- Customizing objects
- Navigating to manage custom objects
Define Project Management Configuration

- Using sandbox

**Important**
You must be assigned the Project Application Administrator role to access the customizable pages. You must create and activate the sandbox for the application.

**Customizing Objects**
The following table summarizes the customizable view objects, pages, and regions in Project Execution Management applications. The view objects listed here are available on the Data Composer user interface for customizations. The Project Application Administrator can then expose the customizations to the corresponding user interface region or pages using the Page Composer.

<table>
<thead>
<tr>
<th>View Object</th>
<th>Application Module</th>
<th>Corresponding Page or Region</th>
<th>Corresponding Business Process</th>
<th>Maximum Custom Attributes Allowed</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequirementVO</td>
<td>RequirementsAM</td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Backlog tab</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>5 text fields including choice list</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Investment Area tab</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Feature tab</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td></td>
</tr>
<tr>
<td>IssueVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Projects, My Work. Expanding the Tasks pane and clicking Manage Issues</td>
<td>Project Management, Control Project, Manage Project Issue</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td>DeliverableSimpleAssociationVO</td>
<td>DeliverablesAM</td>
<td>Navigator, Projects, My Work. Expanding the Tasks pane and clicking Manage Deliverables</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td>IssueTypeVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Tools, Setup and Maintenance. In Tasks pane clicking Manage Task Lists and Tasks. Search for Manage Issue Types task and clicking Go To Task.</td>
<td>Project Management, Setup Project Management, Define Project Management Configuration, Manage Issue Types</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
</tbody>
</table>
### DeliberableTypeVO
- **DeliverablesAM**
  - Navigator, Tools, Setup and Maintenance.
  - In Tasks pane click **Manage Task Lists and Tasks**. Search for Manage Deliverable Types task and click **Go To Task**.
- **Project Management, Setup Project Management, Define Project Management Configuration, Manage Deliverable Types**
  - 30 text fields including choice list, 5 number fields, 5 date time fields

### ResourceRequestVO
- **ResourceRequestAM**
  - ProjectManagerRequestAM
  - Navigator, Projects, Project Management. In the Actions menu drop down click **Manage Project Resources**.
  - Navigator, Projects, Project Management. In the Actions menu drop down click **Manage Project Resource Requests**
- **Project Management, Manage Project Resources, Manage Resource Demand, Create Project Resource Request**
  - 100 text fields, 50 number fields, 20 date fields

---

**Navigating to Manage Custom Objects**

To customize objects, access the Manage Project Execution Management Custom Objects using the Projects Custom Objects link from the Navigator. To set up sandbox, select the Manage Sandboxes menu item from the Administration menu in the global area of Oracle Fusion Applications.

**Using Sandbox**

After you set up a sandbox, you must make it active. The customizations you make are updated in the sandbox. You must publish the sandbox to make the customizations available to other users.

### Define Project Requirement Configuration

**Define Project Requirement Configuration: Overview**

In the Define Project Requirement Configuration implementation tasks, you configure Oracle Fusion Project Management to track requirements, such as features and backlog items, for products and product families.

Setup tasks in the Define Project Requirement Configuration activity are grouped into the following task lists:

<table>
<thead>
<tr>
<th>Task List</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage Project Requirement Levels</td>
<td>A tier within a hierarchy of requirements. For example, a three-level requirement hierarchy may include requirement levels called Investment Area, Feature, and Backlog.</td>
</tr>
</tbody>
</table>
Manage Sprints
A short cycle of development that typically lasts between one and four weeks, in which teams use Agile methodology to develop features for release to consumers. For example, May/2013 and July/2013 represent monthly sprints.

Manage Product Families
A set of shippable products that share characteristics and address complimentary requirements. For example, a software company may have separate product families for HCM and CRM.

Manage Products
Related features bundled into a shippable unit to meet customer requirements. For example, a software company may have a project to develop a Workforce or Compensation Management product.

Manage Releases
Version of a new or enhanced product or product group that is made available to users outside of the development organization. For example, Release 1.0 for a new product or Release 2.1 for a subsequent product release.

Manage Milestones
Milestones are set up as lookup code values to enable you to capture information that is specific to your user story.

Manage Project Requirement Statuses
Requirement statuses are set up as lookup code values to enable you to capture information at the status and task level that is specific to your project.

Manage Custom Attributes for Requirements
Enter any additional attributes for the backlog item. Provide attributes that allow customers enough flexibility to customize their application.

Manage Custom Attributes for Requirements

Customizing Project Execution Management Applications: Explained

Customize view objects for Project Execution Management applications in Oracle Fusion Project Portfolio Management using Data Composer and Page Composer. The following sections describe the aspects of customization:
- Customizing objects
- Navigating to manage custom objects
- Using sandbox

Important
You must be assigned the Project Application Administrator role to access the customizable pages. You must create and activate the sandbox for the application.

Customizing Objects
The following table summarizes the customizable view objects, pages, and regions in Project Execution Management applications. The view objects listed here are available on the Data Composer user interface for customizations. The Project Application Administrator can then expose the customizations to the corresponding user interface region or pages using the Page Composer.
<table>
<thead>
<tr>
<th>View Object</th>
<th>Application Module</th>
<th>Corresponding Page or Region</th>
<th>Corresponding Business Process</th>
<th>Maximum Custom Attributes Allowed</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequirementVO</td>
<td>RequirementsAM</td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Backlog tab</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>5 text fields including choice list</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Investment Area tab</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Feature tab</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IssueVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Projects, My Work.</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Expand the Tasks pane and click Manage Issues</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeliverableSimpleAs</td>
<td>DeliverablesAM</td>
<td>Navigator, Projects, My Work.</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Expand the Tasks pane and click Manage Deliverables</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IssueTypeVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Tools, Setup and Maintenance.</td>
<td>Project Management, Setup Project Management, Define Project Management Configuration, Manage Issue Types</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Tasks pane click Manage Task Lists and Tasks. Search for Manage Issue Types task and click Go To Task.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeliverableTypeVO</td>
<td>DeliverablesAM</td>
<td>Navigator, Tools, Setup and Maintenance.</td>
<td>Project Management, Setup Project Management, Define Project Management Configuration, Manage Deliverable Types</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Tasks pane click Manage Task Lists and Tasks. Search for Manage Deliverable Types task and click Go To Task.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Navigating to Manage Custom Objects

To customize objects, access the Manage Project Execution Management Custom Objects using the Projects Custom Objects link from the Navigator. To set up sandbox, select the Manage Sandboxes menu item from the Administration menu in the global area of Oracle Fusion Applications.

Using Sandbox

After you set up a sandbox, you must make it active. The customizations you make are updated in the sandbox. You must publish the sandbox to make the customizations available to other users.

Manage Project Management 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.
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 Project Management Attachment Categories

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 Manage Project Management Attachment Categories**

**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.

**Integrate with Project Financial Management**

**Oracle Fusion Project Foundation and Oracle Fusion Project Management: How They Work Together**

Use Oracle Fusion Project Integration Gateway to integrate Oracle Fusion Project Foundation with Oracle Fusion Project Management.

The integration enables project accountants, project billing specialists, and executives to centrally perform project costing, billing, accounting, and executive reporting tasks in Oracle Fusion Project Foundation and other Project Financial Management applications while enabling project managers to use Oracle Fusion Project Management to plan, progress, and manage projects.
Oracle Fusion Project Integration Gateway ensures data security, integrity, and efficiency by:

- Enabling you to create an integration for a specific project unit
- Defining a set framework in which data is transferred between Oracle Fusion Project Foundation and Oracle Fusion Project Management

**Managing Integration Options**

Determine how Oracle Fusion Project Foundation and Oracle Fusion Project Management interact. When defining integration options you do the following:

- Specify the integration name and project unit.
- Select an integration planning resource breakdown structure.
- Define how rates are derived when exporting resources.
- Specify how data is processed during import and export.

**Exporting Planning Resources**

Export labor resources and financial resources from the integration planning resource breakdown structure in Oracle Fusion Project Foundation to create project enterprise resources in Oracle Fusion Project Management.

**Exporting Project and Task Information**

Activate integration and export your project and tasks to Oracle Fusion Project Management after you build out the task structure to the required reporting level in Oracle Fusion Project Foundation.

**Restriction**

You cannot export projects if you previously exported them to another scheduling application using Oracle Fusion Project Integration Gateway.

The initial export links project and task information in the two applications, and enables you to incrementally export project and task data to Oracle Fusion Project Management. In Oracle Fusion Project Management, you can add subtasks under the exported tasks. You can also assign resources to the subtasks to complete detailed planning.

**Restriction**

After you export a project, you cannot delete the exported tasks, increase or decrease indent, or move the exported tasks within the project plan in either application.

**Importing Project Plan and Progress Information**

After you export a project, build out the project plan, add subtasks under integrated tasks, assign resources, and complete planning to the desired detail. When you are ready, import the updated project plan and progress information into Oracle Fusion Project Foundation.

When you import a project, the application summarizes planning and progress information for each resource across all subtasks within the hierarchy of each
integrated task. Summarized amounts are imported into a single task assignment created for the resource on the integrated task.

A baseline project plan is automatically created in Oracle Fusion Project Foundation and progress is captured and published. Based on project plan planning options, a baseline budget version can be generated using baseline project plan values and a forecast version generated based on published progress.

**Note**
The current date is always used as the progress as-of date for published progress.

---

**Oracle Fusion Project Management Integration Options: Points to Consider**

Project integration options determine how Oracle Fusion Project Foundation and Oracle Fusion Project Management interact.

When defining integration options you do the following.

- Specify integration definition information including the integration name and project unit.
- Select the integration planning resource breakdown structure.
- Specify resource rate options.
- Specify data processing options.

**Selecting a Project Unit**

You create an integration for a project unit. The project unit you select determines which planning resource breakdown structure you can select as the integration planning resource breakdown structure.

**Selecting a Planning Resource Breakdown Structure**

Select the integration planning resource breakdown structure that is the source of labor resources and financial resources exported to Oracle Fusion Project Management to create project enterprise resources.

**Restriction**
You must select a planning resource breakdown structure that does not allow resource changes at the project level.

All projects exported from Oracle Fusion Project Foundation must use the integration planning resource breakdown structure as the primary planning resource breakdown structure.

**Specifying Resource Rates**

Specify the source of actual cost and bill rates for the resources you export to Oracle Fusion Project Management. The following table describes rate options.
<table>
<thead>
<tr>
<th>Integration Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default business unit</td>
<td>Business unit used to derive the rate schedules that are the source of rates for resources exported to Oracle Fusion Project Management. Determines the currency used to display amounts in Oracle Fusion Project Management after you transfer project and task information. This currency is the primary ledger currency of the business unit.</td>
</tr>
<tr>
<td>Resource class rate schedule</td>
<td>Default source of cost rates when actual rates cannot be derived for the resource.</td>
</tr>
<tr>
<td>Burden schedule</td>
<td>Determines rate calculation if you assign burdened rates to project enterprise resources.</td>
</tr>
<tr>
<td>Effective rate date</td>
<td>Date as of which rates are retrieved and exported.</td>
</tr>
</tbody>
</table>

**Specifying Processing Options**

Specify the number of records to be processed in each set when exporting resources or projects, or when importing project plans from Oracle Fusion Project Management.

**Planning Resources: How They are Exported to Oracle Fusion Project Management**

Export labor resources and financial resources to Oracle Fusion Project Management to enable project managers to plan project tasks using a consistent set of resources and facilitate integration of task assignments with Oracle Fusion Project Foundation.

**Settings That Affect the Export of Planning Resources**

Oracle Fusion Project Management integration options determine how planning resources are exported to create project enterprise resources.

- Integration planning resource breakdown structure: Determines the planning resources you can export.
- Resource rate options: Determine how rates are derived for the planning resources.

The planning resources exported depend on whether you are performing an initial export for your integration or a subsequent one. When you first export resources, all planning resources from the integration planning resource breakdown structure are exported. During subsequent exports, you can select an incremental or a full export, as described in the table below.

<table>
<thead>
<tr>
<th>Processing Type Incremental</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incremental</td>
<td>Export only resources that were added or modified since the last time you successfully exported resources.</td>
</tr>
<tr>
<td>Full</td>
<td>Export all resources from the integration planning resource breakdown structure irrespective of when they were created or modified.</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Tip</td>
<td>Use full export if you want to export rates for all resources (new and existing) for a revised effective rate date in Oracle Fusion Project Management integration options.</td>
</tr>
</tbody>
</table>

**Note**

If you delete a previously exported planning resource from the integrated planning resource breakdown structure, task assignments for the resource are treated as unplanned resources when you next import project plan information from Oracle Fusion Project Management.

**How Planning Resources Are Exported**

As illustrated in the following figure, planning resources are exported to Oracle Fusion Project Management based on resource type.

- Labor resources: If the resource is an HCM person, that is, the Named Person resource type is part of the resource format, then the planning resource is exported as a Person project enterprise labor resource.

- Financial resources: Exported only if the resource type is Expenditure Type.

- Material items and equipment: Not exported.
Define Project Resource Management Configuration

Manage Project Resource Management Implementation Options

Project Resource Management Workflow Notifications: Explained

Workflow notifications perform the following actions in Oracle Fusion Project Resource Management:

- Notify the staffing owner when a requester submits a project resource request for fulfillment. This notification is for information only.

- Notify the requester of a project resource request when a resource is proposed to fulfill the assignment. The requester must approve or reject the proposed resource. The workflow notification process starts when the staffing owner assigns a resource to fulfill a project resource request. The requester can approve or reject the proposed resource by using any of these methods:
  - Approve or Reject buttons in the notification in the Worklist: Notifications and Approvals region
  - Approve or Reject links in the e-mail notification
  - Actions menu on the project resource request

- Notify the staffing owner when the requester approves or rejects the proposed resource for the assignment. This notification is for information only.

- Notify the resource pool owner when a resource is no longer eligible to fulfill project resource requests. If you deselect the option to manage resource availability and staffing for a project enterprise labor resource, then the resource is no longer managed in Oracle Fusion Project Resource Management. The application sends a notification to the owner of the resource pool to which the resource belongs. The notification is for information only.
Project Resource Management Notification Settings

You must select the Notify requester when a resource is proposed to fulfill a project resource request option on the Manage Project Resource Management Implementation Options page to enable workflow notifications when a resource is proposed, approved, or rejected for a project resource request.

Note
If you do not enable the workflow implementation option, then the requester can approve or reject the resource on the project resource request.

Workflow notifications are automatically enabled in the application to notify a resource pool owner when a resource is no longer eligible to fulfill project resource requests. No implementation setup is required for this type of workflow.

Project Resource Request Workflow Notification Example
The following figure shows an example project resource request flow that uses workflow notifications.

In this example, a project manager submits a request for a new project resource. The resource manager searches for qualified, available candidates and finds a resource that is a good match for the project. The resource manager specifies the resource name on the request and submits the request for the project manager to approve the proposed resource. The project manager receives an e-mail notification requesting approval of the resource, and approves the resource. This
action fulfills the project resource request, assigns the resource to the project, and launches an e-mail notification to inform the resource manager that the resource is approved.

Define Attributes for Employees and Contingent Workers in Human Capital Management

Manage Legislative Data Groups

Legislative Data Groups: Explained

Legislative data groups are 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.

Legislative Data Groups

Oracle Fusion Payroll is organized by legislative data groups. Each legislative data group marks a legislation in which payroll is processed, and is associated with a legislative code, currency and its own cost key flexfield structure. A legislative data group is a boundary that can share the same set up and still comply with the local laws. It can span many jurisdictions as long as they are within one country, and contain many legal entities that act as payroll statutory units. Each payroll statutory unit can belong to only one legislative data group.

Manage Business Unit

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.

**Manage Legal Addresses**

**Creating Legal Jurisdictions, Addresses and Authorities: Examples**

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

**Legal Jurisdictions**

Create a legal jurisdiction by following these steps:

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

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

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

### Legal Authorities
Create a legal authority by following these steps:

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

**Note**
Create an address for the legal authority.

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

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

17. Select **Add Row**.
18. Select **Purpose**.
19. The **Purpose from Date** will default to today's date.
20. Optionally enter a **Purpose to Date**.
21. Select **OK**.
22. Select **Save and Close**.

**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...
Define Project Resource Management Configuration

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.
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.
Define Project Resource Management Configuration

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.

**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.

Manage Job

Jobs: Explained

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

Basic Details

Basic details for a job include an effective start date, a job set, a name, and a code.

A job code must be unique within a set. Therefore, you can create a job with the code DEV01 in the US set and another job with the same code in the UK set. However, if you create a job with the code DEV01 in the Common set, then you cannot create a job with the same code in any other set.

Benchmark Information

You can identify a job as being a benchmark job. A benchmark job represents other jobs in reports and salary surveys. You can also select the benchmark for
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.
Define Project Enterprise Labor Resources

Project Enterprise Labor Resource Components: How They Work Together

A project enterprise labor resource is a resource that you can assign to multiple projects. If you manage resource availability and staffing in Oracle Fusion Project Resource Management, use project enterprise labor resources to fulfill project resource requests.

Consider the following attributes when creating a project enterprise labor resource.

- Select the type attribute
- Select the resource calendar
- Select the primary project role
- Associate an employee or contingent worker
- Manage resource availability and staffing
- Select a resource pool
- Manage rates and amounts

Selecting the Type Attribute

You can add both labor and expense type resources to a project. Project enterprise labor resources are the Labor resource type.

Selecting the Resource Calendar

Resource calendars are used to determine resource availability, schedule tasks, and assign resources to tasks.

The main purpose of the calendar is for daily scheduling of work using hours per day as the unit. The duration of tasks is measured in days and not hours.
A predefined calendar with eight hours per day and five days per week with no holidays is used as a default if no calendar is available for a project or resource.

**Selecting the Primary Project Role**

Available values are based on Oracle Fusion Project Management project roles. The primary project role represents the role that a resource most often fulfills on a project.

**Note**

The default value is Project Team Member.

**Associating an Employee or Contingent Worker**

An HCM person is a named person in the HCM database with a person type of Employee or Contingent Worker and an active assignment. In Oracle Fusion Project Resource Management, if you associate an HCM person with a project enterprise labor resource and manage the resource's availability and staffing, then the resource is eligible to fulfill project resource requests.

**Managing Resource Availability and Staffing**

Select this option for the resource to be eligible to fulfill project resource requests. If you deselect this option after selecting it, the resource availability and staffing is no longer managed in Oracle Fusion Project Resource Management, and the following changes occur:

- If the resource is not used in Project Resource Management, then the resource is no longer eligible to fulfill project resource requests and is removed completely from pools.
- If the resource is used in Project Resource Management, then resource is no longer eligible to fulfill project resource requests and remains in pools for historical reporting purposes. If the resource is already associated with an unfulfilled request, then the resource is no longer eligible to be assigned to fulfill the request.
- The application sends the resource pool owner a notification that the resource is no longer eligible to fulfill project resource requests. This notification is for information only.

**Selecting a Resource Pool**

You must select a resource pool if you manage the resource availability and staffing in Project Resource Management.

**Note**

The default value is the Resources with No Pool Membership pool.

**Managing Rates and Amounts**

The rate and amount types on a project are:
• Bill Rate: The rate for a unit of work that determines the invoice or revenue recognized amount for a resource on a project.

• Cost Rate: The rate for a unit of work that determines the cost for a resource on a project.

• Bill Amount: The bill amount for a resource allocated to a project, calculated by multiplying the labor effort by the project resource bill rate.

• Cost Amount: The cost amount for a resource allocated to a project, calculated by multiplying the labor effort by the project resource cost rate.

You can assign cost and bill rates to a resource on the project enterprise labor resource definition. When you add a resource to a project, the application copies the resource cost and bill rates to the project if the rate currency is the same as the project currency.

You can edit planning-only resource rates directly on the project, and the edits do not affect the rates on the resource definition. You cannot edit the rates on the project for resources in an Assigned status.

Manage Project Enterprise Labor Resource Maintenance Conditions

Project Enterprise Labor Resources: How They Are Automatically Created and Maintained

Run the Maintain Project Enterprise Labor Resources process to create resources, remove resource availability for staffing, and maintain the resource search index.

Settings That Affect the Maintain Project Enterprise Labor Resources Process

Create and maintain project enterprise labor resources by defining a set of conditions that specify the employees and contingent workers in Oracle Fusion Human Capital Management (HCM) to create as resources in Oracle Fusion Project Portfolio Management. Define the resource attribute values, such as the resource calendar and primary project role, to assign to the new resources.

The following table lists the criteria that you define for selecting HCM persons to create as project enterprise labor resources.

<table>
<thead>
<tr>
<th>Selection Criteria</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person type</td>
<td>Select the HCM person type for the process to use as criteria when selecting HCM persons to create as resources. Available person types are employees, contingent workers, or both employees and contingent workers.</td>
</tr>
<tr>
<td>Option to select people based on job</td>
<td>Select this option for the process to create resources for HCM persons with specific jobs.</td>
</tr>
</tbody>
</table>

Caution

If you do not select job criteria for this condition, the process will create resources for all eligible people based only on the person type.
If you selected the option to select people based on job, then select the jobs for the process to use as HCM person selection criteria.

<table>
<thead>
<tr>
<th>Resource Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option to manage resource availability and staffing</td>
<td>Select this option for resource managers to manage the availability and project staffing of resources who are created based on the condition. Resources who are managed in Oracle Fusion Project Resource Management are eligible to fulfill project resource requests.</td>
</tr>
<tr>
<td>Calendar</td>
<td>Select the resource calendar for resources that are created based on the condition. The resource calendar defines the work schedule for project enterprise labor resources, and includes working days and hours and any exceptions, such as holidays.</td>
</tr>
<tr>
<td>Primary project role</td>
<td>Select the primary project role for resources that are created based on the condition. Project roles are maintained in Oracle Fusion Project Management.</td>
</tr>
<tr>
<td>Cost rate and currency</td>
<td>Enter the cost rate and currency for resources that are created based on the condition.</td>
</tr>
<tr>
<td>Bill rate and currency</td>
<td>Enter the bill rate and currency for resources that are created based on the condition.</td>
</tr>
</tbody>
</table>

**How Resources are Created**

Select one or both of the following process options when you run the Maintain Project Enterprise Labor Resources process:

- **Create resources:**
  Creates project enterprise labor resources from Oracle Fusion HCM persons based on person selection criteria and resource attribute values defined in maintenance conditions.

- **Remove resource availability:**
  Disables the option to manage resource availability and staffing for all project enterprise labor resources with no active primary assignment in Oracle Fusion HCM.

The Maintain Project Enterprise Labor Resources process includes all conditions on the Maintain Project Enterprise Labor Resources Process Conditions page each time the process runs. Assign a process order to each condition to determine the order in which the Maintain Project Enterprise Labor Resources process creates resources. When you run the process for the Create Resources option, conditions are processed in ascending order of the process order value. Multiple conditions with the same process order value are processed in ascending alphabetic order of the condition name.

The Maintain Project Enterprise Labor Resources process creates resources only for HCM people that meet the selection criteria for a condition and are
Define Project Resource Management Configuration

not already linked to a project enterprise labor resource. The list of exceptions appears in the Last Process Details region and refreshes each time you run the process.

Resources who no longer have the option enabled to manage resource availability and staffing are not eligible to appear in match results. You have the option to disable resource availability for resources who do not have an active primary assignment in HCM. The application sends resource pool owners a notification when resources in their pools are no longer eligible to fulfill project resource requests. The pool owner, who is typically the resource manager, may need to take further action after reviewing the notification. For example, if a resource is currently on a project assignment, but is no longer eligible for future assignments, then the resource manager may need to adjust the finish date of the resource's current assignment. The resource may agree to complete the assignment as scheduled, or the assignment might end immediately if the resource left the company. The resource manager and project manager typically work together in these circumstances to determine the best course of action for the duration of the assignment.

After all conditions are processed, the Maintain Project Resource Search Index Job Set process runs. This process updates the index that enables resources to appear on the Search and Evaluate Resources page when searching for resources to fulfill project resource request. You can also submit the job set for the index on its own. It is not necessary to always run the maintain process first.

Manage Resource Pools

Resource Pools: Explained

A resource pool is a logical group of resources organized in a hierarchy for purposes of staffing and managing resources, and reporting on utilization.

Creating and Deleting Resource Pools

Use the Manage Resource Pools page to build a resource pool hierarchy that reflects how the organization manages and reports on resources. For example, resource pools can group resources in a hierarchy by geographic locations, cost centers, skill sets, or primary project roles.

The pools named All Resources and Resources with No Pool Memberships are predefined. The All Resources pool is always at the top of the resource pool hierarchy. The Resources with No Pool Memberships pool is a direct child of the All Resources pool.

You can revise the resource pool hierarchy, such as moving a child pool from a parent pool in one hierarchy to a parent pool in a different hierarchy. However, you cannot move or delete the All Resources pool or Resources with No Pool Membership pool.

You can assign any project enterprise labor resource as a resource pool owner. Typically resource managers own the resource pools. When a resource manager searches for resources to fulfill a request, by default the application searches for resources who are members of resources pools that the resource manager owns.
The search also includes any resource pools that are children of the resource pools that the resource manager owns.

You must remove all resources that are members of a pool before you can delete that pool. You must remove all resources that are members of child pools of a parent pool before you can delete the parent pool. If you delete a parent pool, the child pools are also deleted.

**Maintaining Resource Pool Membership**

All managed project enterprise labor resources are members of a resource pool. Resources can belong to only one pool. The **Resources with No Pool Memberships** pool is the default pool for new resources.

You can move resources to different resource pools individually or as a group. Use the Manage Resource Pools page to change the resource pool that resources belong to. You can select the source pool on the resource pool hierarchy and then drag and drop pool members onto the target pool. You can also move resources by selecting pool members and clicking the **Move Resources** button to open a window to select the target resource pool.

If you deselect the option to manage a resource's availability and staffing in Oracle Fusion Project Resource Management, then the resource is no longer eligible to fulfill project resource requests. If you deselect this option after selecting it, the resource's pool membership is affected as follows:

- If the resource is not used in Oracle Fusion Project Resource Management, then the application automatically removes the resource from any resource pool.
- If the resource's name appears in the Resource field on a project resource request in any status, then the resource remains in a resource pool for historical reporting purposes.

**Using Resource Pools**

When resource managers search for resources to fulfill project resource requests, they can search from pools that they own, or select one or more resource pools to focus the search to specific groups of resources. There is no restriction to the resource pools that resource managers can search to find qualified resources to fulfill project resource requests.

Resource managers use the resource pool hierarchy to review projected utilization on the Resource Manager Dashboard. The dashboard monitors utilization of resources and resource pools and other related analytics.

**Submit Process to Maintain Project Resource Search**

**Project Resource Search Index: How It Is Maintained**

The Maintain Project Resource Search Index job set consists of processes that facilitate efficient and rapid searching for qualified resources to fulfill project resource requests.
The job set maintains the search index of project resource qualifications, qualification proficiencies, and filters. The application uses the index to calculate resource Qualification scores and filter counts during the search for resources, and to display resource details.

These actions depend on indexed resource data:

- Initiating a search for resources for a project resource request
- Modifying search criteria on the Search and Evaluate Resources page and conducting a new search
- Modifying the resource pools that the search uses
- Increasing the resource match thresholds for the Qualification or Available Capacity scores
- Selecting new search filters, such as competencies, languages, locations, and project roles
- Displaying resource details

**How the Project Resource Search Index Is Maintained**

You can run or schedule the Maintain Project Resource Search Index job set on the Scheduled Processes page. In addition, the Maintain Project Resource Search Index job set starts automatically after you run the Maintain Project Enterprise Labor Resources process to create resources or remove resource availability for staffing.

The Maintain Project Resource Search Index job set starts the following processes to update the search index.

- **Maintain Project Resource Qualification Index process**
  Maintains the index of active qualifications and qualification proficiencies for project resources. A qualification is indexed if the date the process runs is within the effective dates of the qualification. The index is used to calculate project resource qualification scores when searching for qualified resources to fulfill a project resource request.

- **Maintain Project Resource Qualification Filters process**
  Maintains the qualification filters and count of resources that have each qualification.

- **Maintain Project Resource Single Attribute Filters process**
  Maintains the filters for attributes for which a resource can have only a single value, such as location and project role. Maintains the count of resources that have each attribute.

Run the Maintain Project Resource Search Index job set often enough to account for the following changes:

- Changes in qualifications and proficiencies for an HCM person who is associated with a project enterprise labor resource.
- Enabling a resource’s eligibility to fulfill project resource requests, which is indicated on the resource profile with the option to manage the resource availability and project staffing.
• New or terminated resources.

For example, assume that you require resources to update their qualifications when they complete a project, and the average project duration is two weeks. You can schedule the Maintain Project Resource Search Index job set to run once every two weeks to update the index with the latest resource qualification changes.

**Important**

Resource managers cannot search for resources or view resource details if the search index is not available. The search index will not be available if the Maintain Project Resource Search Index process fails, or during the time that the process is running.

**FAQs for Define Project Enterprise Labor Resources**

**What’s the difference between a project enterprise labor resource and a managed project enterprise labor resource?**

A project enterprise labor resource is a resource that you can assign to multiple projects.

A managed project enterprise labor resource is a project enterprise resource who is associated with a named person in HCM, and whose availability and staffing are managed in Oracle Fusion Project Resource Management. You use managed project enterprise labor resources to fulfill project resource requests.

**Who can be a member of a resource pool?**

To be a member of a resource pool, you must be a managed project enterprise labor resource whose availability and staffing are managed in Oracle Fusion Project Resource Management.

A resource who was previously eligible for staffing and fulfilled a project resource request will remain a member of the pool even if the Manage Resource Availability and Staffing option is subsequently disabled for the resource.

**Can I delete a resource from a resource pool?**

No. Instead, you can move the resource to another resource pool. For example, you can create a pool for inactive resources.

**How can I change the resource pool for resources?**

To change the resource pool that resources belong to, use the Manage Resource Pools page. Select the source pool on the resource pool hierarchy and then select, drag, and drop pool members onto the target pool in the resource pool hierarchy.
Alternatively, you can select pool members and click the **Move Resources** button to open a window to select the target resource pool.

**What is the processing order of project enterprise labor resource maintenance conditions?**

Records are processed by the Maintain Project Enterprise Labor Resources process in ascending order based on the process order value that you specify on the Create Condition page. For example, assume that an HCM person is included in a condition with a process order value of 1. If a resource does not already exist for that HCM person, then the process creates a resource for the condition with the process order value of 1, even if subsequent conditions also apply to the HCM person.

If multiple conditions have the same process order value, then records are processed in ascending alphabetic order based on the condition name.

**How can I diagnose issues with resource eligibility when creating project enterprise labor resources?**

Run the Project Enterprise Labor Resource Eligibility diagnostic test from the Help - Troubleshooting - Run Diagnostic Tests menu to determine whether a person in Oracle Fusion HCM is eligible to become a project enterprise labor resource. The test generates a report showing the results of three eligibility parameters.

- **Project Enterprise Labor Resource**: Validates that a project enterprise labor resource does not exist for the person.
- **Work E-Mail**: Validates that the person has a valid work e-mail, and the e-mail is not associated with an existing project enterprise labor resource or a terminated HCM person.
- **Primary Assignment**: Validates that the person has an active primary assignment for a person type of Employee or Contingent Worker.

**Manage Target Utilization Percentages**

**Resource Target Utilization Percentage: How It Is Determined**

Target utilization is the percentage of hours that project enterprise labor resources are expected to work on project assignments compared to their available hours. The project administrator specifies the default target utilization for the enterprise, and then specifies target utilization overrides for individual jobs.

**Factors That Affect Resource Target Utilization Percentage**

The following factors affect the Resource Target Utilization Percentage.
• Enterprise Target Utilization Percentage: Target utilization percentage that applies to all managed project enterprise labor resources in the enterprise unless specific override percentages exist for a resource’s job.

• Target Utilization Percentage Overrides: Target utilization percentages that override the enterprise target utilization percentage for resources with specific jobs.

**How the Resource Target Utilization Percentage Is Determined**

The application uses the enterprise target utilization percentage for a resource if no override percentage exists for the resource’s job. The default target utilization is 100%. The project application administrator can enter a new value in the Enterprise Target Utilization Percentage field on the Manage Target Utilization Percentages page.

If you enter a target utilization percentage override on the Manage Target Utilization Percentages page for a resource’s job, then the application uses that percentage to compare to the resource’s projected utilization on the Resource Manager Dashboard. The job is based on the resource’s active primary HCM assignment.

Target utilization percentages can be whole numbers from 0 - 100. You can enter one override percentage for each job.

**Example**

Assume that the target utilization for the enterprise is 80%. To set up the target utilization, change the default enterprise target utilization percentage from 100% to 80% on the Manage Target Utilization Percentages page.

Also assume that Senior Architects spend half their time on internal activities. To set up the target utilization for Senior Architects, enter 50% as the target utilization override for the job of Senior Architect.

The application will use 50% as the target utilization for all project enterprise labor resources with the job of Senior Architect, and 80% utilization for all other project enterprise labor resources.

**Manage Custom Attributes for Project Resource Requests**

**Customizing Project Execution Management Applications: Explained**

Customize view objects for Project Execution Management applications in Oracle Fusion Project Portfolio Management using Data Composer and Page Composer. The following sections describe the aspects of customization:

- Customizing objects
- Navigating to manage custom objects
- Using sandbox
**Important**

You must be assigned the Project Application Administrator role to access the customizable pages. You must create and activate the sandbox for the application.

**Customizing Objects**

The following table summarizes the customizable view objects, pages, and regions in Project Execution Management applications. The view objects listed here are available on the Data Composer user interface for customizations. The Project Application Administrator can then expose the customizations to the corresponding user interface region or pages using the Page Composer.

<table>
<thead>
<tr>
<th>View Object</th>
<th>Application Module</th>
<th>Corresponding Page or Region</th>
<th>Corresponding Business Process</th>
<th>Maximum Custom Attributes Allowed</th>
</tr>
</thead>
<tbody>
<tr>
<td>RequirementVO</td>
<td>RequirementsAM</td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Backlog tab</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>5 text fields including choice list</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Investment Area tab</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Navigator, Projects, Requirements, Manage Requirements page, Feature tab</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IssueVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Projects, My Work. Expand the Tasks pane and click Manage Issues</td>
<td>Project Management, Control Project, Define Project Issue</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td>DeliverableSimpleAssociationVO</td>
<td>DeliverablesAM</td>
<td>Navigator, Projects, My Work. Expand the Tasks pane and click Manage Deliverables</td>
<td>Project Management, Plan Project, Define Project Scope and Deliverables</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
<tr>
<td>IssueTypeVO</td>
<td>IssueManagementAM</td>
<td>Navigator, Tools, Setup and Maintenance. In Tasks pane click Manage Task Lists and Tasks. Search for Manage Issue Types task and click Go To Task.</td>
<td>Project Management, Setup Project Management, Define Project Management Configuration, Manage Issue Types</td>
<td>30 text fields including choice list, 5 number fields, 5 date time fields</td>
</tr>
</tbody>
</table>
Navigating to Manage Custom Objects

To customize objects, access the Manage Project Execution Management Custom Objects using the Projects Custom Objects link from the Navigator. To set up sandbox, select the Manage Sandboxes menu item from the Administration menu in the global area of Oracle Fusion Applications.

Using Sandbox

After you set up a sandbox, you must make it active. The customizations you make are updated in the sandbox. You must publish the sandbox to make the customizations available to other users.

Manage Project Resource Management 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.

Manage Project Resource Management Attachment Categories

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 Manage Project Resource Management Attachment Categories**

**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.
### Other Common Setup and Maintenance Tasks

#### Define Transactional Business Intelligence Configuration

**Define Transactional Business Intelligence Configuration: Highlights**

Configure Oracle Transactional Business Intelligence for ad hoc reporting, review certain setup objects to be used in Transactional Business Intelligence, and manage the presentation catalog and currency type display.

**Defining Transactional Business Intelligence Configuration**

- Review details about the Transactional Business Intelligence tasks. Refer to the Oracle Fusion Transactional Business Intelligence Administrator’s Guide.

#### Access to Person Data

**Assigning Security Profiles to Job Roles for Oracle Fusion Transactional Business Intelligence Users: Explained**

Users of Oracle Fusion Transactional Business Intelligence (Transactional Business Intelligence) need access to some person data for reporting purposes. To provide this access, you assign a predefined security profile to relevant job or abstract roles using the Oracle Fusion Human Capital Management (HCM) setup task Manage Data Role and Security Profiles. On completion of this task, Oracle Fusion Data Security is updated automatically for roles being used to access Transactional Business Intelligence.

**Job or Abstract Roles and Related Security Profiles**

The following table identifies, by Oracle Fusion product, the job and abstract roles that need access to person data and the predefined security profile that you assign to each role.

<table>
<thead>
<tr>
<th>Product</th>
<th>Job or Abstract Role</th>
<th>Security Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Fusion Assets</td>
<td>Asset Accountant</td>
<td>View All Workers</td>
</tr>
</tbody>
</table>
For example, as part of their Transactional Business Intelligence setup:

- Oracle Fusion Assets implementors must assign the predefined security profile View All Workers to the Asset Accountant and Asset Accounting Manager job roles.

- Oracle Fusion Incentive Compensation implementors must assign the predefined security profile View Manager Hierarchy to the abstract role Incentive Compensation Participant Manager.

The security profiles that HCM roles need to access Transactional Business Intelligence are assigned during the setup of HCM data security: no additional setup is required for Transactional Business Intelligence purposes.

**Enabling an Oracle Fusion Transactional Business Intelligence User to Access Person Data: Worked Example**

This example shows how to assign a security profile to a job or abstract role to enable users with that role to access person data. This task is required for users of Oracle Fusion Transactional Business Intelligence (Transactional Business Intelligence) who do not also use Oracle Fusion Human Capital Management (HCM).

The following table summarizes key decisions for this scenario. When performing this task, use the job or abstract role for your product and the name of the relevant predefined person security profile in place of those shown here.

<table>
<thead>
<tr>
<th>Application</th>
<th>Role</th>
<th>Security Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Fusion Assets</td>
<td>Asset Accounting Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Enterprise Planning and Budgeting</td>
<td>Budget Analyst</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Enterprise Planning and Budgeting</td>
<td>Budget Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financial Consolidation Hub</td>
<td>Consolidation Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financial Consolidation Hub</td>
<td>Consolidation Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Financials Common Module</td>
<td>Intercompany Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>Financial Analyst</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>General Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion General Ledger</td>
<td>General Accounting Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Incentive Compensation</td>
<td>Incentive Compensation Participant Manager</td>
<td>View Manager Hierarchy</td>
</tr>
<tr>
<td>Oracle Fusion Inventory Management</td>
<td>Warehouse Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Accountant</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Administrator</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Project Foundation</td>
<td>Project Billing Specialist</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Purchasing</td>
<td>Buyer</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Sourcing</td>
<td>Category Manager</td>
<td>View All Workers</td>
</tr>
<tr>
<td>Oracle Fusion Sourcing</td>
<td>Sourcing Project Collaborator</td>
<td>View All Workers</td>
</tr>
</tbody>
</table>
### Decisions to Consider

| What is the name of the Transactional Business Intelligence job or abstract role? | Warehouse Manager |
| What is the name of the person security profile? | View All Workers |

### Summary of the Tasks

To perform these tasks, you must have the role IT Security Manager.

1. Launch the task Manage Data Role and Security Profiles.
2. Search for the job or abstract role.
3. Assign the relevant predefined security profile to the job or abstract role.

#### Launching the Task Manage Data Role and Security Profiles

1. On the Overview page of the Setup and Maintenance work area, click the **All Tasks** tab.
2. In the Search region, complete the fields as shown in this table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Tasks</td>
</tr>
<tr>
<td>Name</td>
<td>Manage Data Role and Security Profiles</td>
</tr>
</tbody>
</table>

3. Click **Search**.
4. In the search results, click **Go to Task** for the Manage Data Role and Security Profiles task.

#### Searching for the Job or Abstract Role

1. On the Manage HCM Data Roles page, enter the job name Warehouse Manager in the **Role** field.
2. Click **Search**.
3. In the search results, highlight the entry for the Warehouse Manager job role.
4. Click **Assign**.

#### Assigning the Security Profile to the Job Role

1. In the **Person Security Profile** field on the Assign Data Role: Security Criteria page, select the security profile View All Workers.
2. Click **Review**.
3. On the Assign Data Role: Review page, click **Submit**.
Define Extensions: Define Custom Enterprise Scheduler Jobs for
Project Execution Management

Managing Job Definitions: Highlights

Oracle Enterprise Scheduler jobs are run in Oracle Fusion Applications to
process data and, in some cases, to provide report output. A job definition
contains the metadata that determines what the job does and what options are
available to users who run the job. You can create and maintain job definitions
for use in Oracle Fusion Applications.

Managing job definitions is described in the Oracle Fusion Applications
Administrator’s Guide and Oracle Fusion Applications Extensibility Guide
for Developers. As you read content from these guides, note that the guides
mention managing Oracle Enterprise Scheduler, including job definitions, from
Oracle Enterprise Manager Fusion Applications Control. You can also access
job definitions by starting in the Setup and Maintenance Overview page and
searching for the Enterprise Scheduler job tasks for your applications.

Note

Oracle Enterprise Manager Fusion Applications Control is not available for
Oracle Cloud implementations.

Selecting the Appropriate Implementation Task

Each Enterprise Scheduler job definition task uses one Java EE application,
which is referenced in the task name. You must use the right task because, to
access the product job definition that you want to view or work on, the view
objects must be contained in the application. If you do not select the right task,
then the job definition will not be displayed properly or function correctly.
The application name is usually the same as the product that the job definition
belongs to, but not always.

- For example, the Oracle Fusion Payables Java EE application contains the
Oracle Fusion Expenses product view objects. To create or maintain a job
definition for use in Expenses, you select the Manage Custom Enterprise
Scheduler Jobs for Payables and Related Applications task.

- In another example, the Oracle Fusion Payments product view objects are
contained in both Oracle Fusion Payables and Oracle Fusion Receivables
Java EE applications. You need to select the task appropriate to the job
definition for Payments. Use the Manage Custom Enterprise Scheduler
Jobs for Receivables and Related Applications task if the job is for
receivables functionality, or the Manage Custom Enterprise Scheduler
Jobs for Payables and Related Applications task if the job is for payables
functionality.

- Use the task description to see the products that correspond to the Java
EE application specified in the task name. For example, the description
for the Payables task lists Oracle Fusion Payables, Assets, Expenses, and Payments.

- You can view task descriptions in the help window for the task, if any, or in the generated setup task lists and tasks report from the Getting Started page.
- If you have access to the Manage Task Lists and Tasks page, you can also open the details for specific tasks to see the description.
- For general information about product and Java EE application relationships, use Oracle Enterprise Manager Fusion Applications Control (Fusion Applications Control).

See: Topology Section

Viewing, Creating, and Editing Job Definitions

- You can access predefined and custom job definitions. In the table on the Manage Job Definitions tab, the Name column displays an asterisk for predefined job definitions. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Viewing Job Definitions

- You or a technical administrator can create jobs based on Java, PL/SQL, Oracle Business Intelligence (BI) Publisher, or any other supported technology. Every predefined or custom job must have a job definition. For Oracle Cloud implementations, custom job definitions can be created only for custom jobs based on Oracle BI Publisher reports. Refer to the Oracle Fusion Applications Administrator's Guide.

See: Creating a Job Definition

- If you are using the Setup and Maintenance work area, then the Enable submission from Enterprise Manager check box is available for the job definition.
  - If you do not select this check box, then the job cannot be run from Enterprise Manager.
  - If you select this check box, then you can define parameters for this job definition only in Enterprise Manager. Save the rest of your work on the job definition, and then go to Enterprise Manager if you need to define parameters.

- You can edit all aspects of custom job definitions. For predefined job definitions, you can’t update parameters, but you can determine if user properties are read-only or not. You can also edit certain aspects of predefined definitions, which are described as job properties in the Oracle Fusion Applications Extensibility Guide for Developers.

See: Customizing Existing Oracle Enterprise Scheduler Job Properties

Managing List of Values Sources: Highlights

A list of values source for Oracle Enterprise Scheduler job definitions determines where a list of values comes from and what the specific values are. These lists
of values are used in parameters and application defined properties of job definitions. For example, you can use a source of country values for a Country job parameter.

Note

Since parameters for predefined job definitions cannot be edited, list of values sources are only for parameters in custom job definitions.

Managing list of values sources is fully described in the Oracle Fusion Applications Administrator’s Guide. As you read content from that guide, note that the guide describes managing Oracle Enterprise Scheduler, including list of values sources, from Oracle Enterprise Manager Fusion Applications Control. You can also access list of values sources by starting in the Setup and Maintenance Overview page and searching for Enterprise Scheduler job tasks.

Registering and Searching for List of Values Sources

- Create list of values sources to register them for use in job definitions.

  See: Registering Sources for Lists of Values

- Search for list of values sources to edit or delete, or to make sure a particular source does not already exist before you create it.

  See: Searching for List of Value Sources

Contextual Addresses

Setting Up the Mapping Service for Contextual Addresses: Points to Consider

A contextual address is marked with an orange square contextual action icon that can be clicked to display the address on a map. Use the Mapping Service for Contextual Addresses profile option to specify the web mapping service to be used to display the map. In the Setup and Maintenance work area, go to the Manage Administrator Profile Values task to set the profile option value.

Profile Option Default

By default, the Mapping Service for Contextual Addresses profile option has no value. Until you enter a valid value for this profile option, users get an error when they try to open a map for any contextual address.

Profile Option Value

After you find and select the Use the Mapping Service for Contextual Addresses profile option, enter a mapping service URL in the Profile Value column, for example:

• http://maps.yahoo.com/maps_result.php?q1=
• http://maps.live.com/default.aspx?where1=
• http://bing.com/maps/?v=2&encType=1&where1=

Optionally, add parameters to the URL. For example, to avoid a locator box in Google Maps, add &iwloc=& to the URL, so that you enter http://maps.google.com/maps?iwloc=&&output=embed&q= as the profile value.
Web Services: Overview

Use web services to integrate web-based applications into your Oracle Fusion applications. Web services expose Oracle Fusion Applications business objects and processes to other applications through the use of open standards-based technologies. Some of these technologies include Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Business Process Execution Language (BPEL), Web Services Description Language (WSDL), and XML schema definitions (XSD). Oracle Fusion Applications web services support development environments and clients that comply with these open standards.

Oracle Fusion Applications includes two types of web services: Application Development Framework (ADF) services and composite services. The following table describes the two types.

<table>
<thead>
<tr>
<th>Web Service Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| ADF services     | ADF services usually represent business objects, such as employees or purchase orders. ADF services typically expose standard operations, such as create, update, and delete. However, for locally-persisted objects, ADF services are not limited to these operations. Examples of ADF services include:  
  - Worker.changeHireDate - a service that updates the hire date of the worker business object.  
  - ProjectTask.createTask - a service that adds a task to the project task business object. |

|
Composite services usually represent end-to-end business process flows that act on business events produced by the ADF services. Composite services orchestrate multiple object-based services, rules services, and human workflows. Examples of composite services include:

- `ProjectStatusChangeApproval.process` - a service that accepts the change in project status.
- `ScheduleOrchestrationOrderFulfillmentLineService.scheduleOrders` - a service that schedules resources used to fulfill an order.

Access Oracle Enterprise Repository for Oracle Fusion Applications to find detailed information about integration assets, such as web services. To view lists of web services, select these asset types:

- ADF Service
- ADF Service Data Object
- Composite Service
- Composite

Service methods and parameters, the service path, the WSDL URL and other technical data, appear on the Detail tab of each web service. Step-by-step instructions regarding the invocation of a service and the service XSD appear on the Documentation tab.

**Files for Import and Export**

**Files for Import and Export: Explained**

You can import data into or export data out of Oracle Fusion Applications using repositories of content and processes for import and export.

Integration specialists stage data for import and export. Application administrators run processes to import data in repositories of content to application transaction tables, or retrieve data exported from applications.

Aspects of managing files for import and export involve the following:

- The File Import and Export page
- Interacting with content management
- Uploading for import
- Downloading for export
- File size
The File Import and Export Page

The File Import and Export page lets you upload content to or download content from the document repository of Oracle WebCenter Content Management. For information or assistance regarding general access to content management (including all metadata), to create and manage accounts, and to programmatically upload and download content, contact the WebCenter Content Administrator.

Search criteria on the page are limited to the minimum metadata of content management records needed for file import and export.


Interacting with Content Management

Everyone who uses the File Import and Export page is assigned to one or more accounts in content management.

Accounts organize and secure access to content items.

Uploading for Import

Uploading a file creates a record.

When you create a record, you must specify an account as well as the file. When you create a record, you must specify an account as well as the file. The account you specify determines which import process picks up that file to import it.

You can upload any file formats that can be parsed by the content repository being used, such as any MIME or content types. However, the format uploaded should conform to the requirements of the import process being used, such as a comma-separated values (CSV) file for the Load Interface File for Import process.

Downloading for Export

Processes you run to export data result in files in content management. Records in the search results table of the File Import and Export page provide links to the files for download.

Note

The owner of a data export file can be an application ID (APPID).

File Size

Upload and download does not intentionally apply the following:

- Data compression
- File chunking or splitting
The `UPLOAD_MAX_DISK_SPACE` parameter in the `web.xml` file determines the maximum allowable file size in content management. The default maximum size is 1024000 (10MB).

**Files for Import and Export: Points to Consider**

Interaction between the File Import and Export page and Oracle WebCenter Content requires securing content in an account. Oracle provides predefined accounts in Oracle WebCenter Content.

Areas of file import and export to consider involve the following:

- Security
- Searching records
- Accessing content in a new account
- Account names
- Deleting files

**Security**

The duty role needed for accessing the File Import and Export page is File Import and Export Management duty. This duty role is included in the predefined role hierarchy for integration specialist roles and product family administrator roles.

Files in Oracle WebCenter Content are associated with an account so that only users who have permission to a particular account can work with content items that belong to that account. You can only upload and download files to and from content management that are associated with accounts that you are entitled to access.

Oracle WebCenter Content does not support trailing slashes (/). Account names are appended with a $ to ensure each account is unique. Account names are dynamic so that if they overlap (one name is completely contained in another, longer name, such as US and USSales), each account is treated as discrete by access grants.

Security such as virus scanning is handled by the underlying integrated content management.

**Searching Records**

A record in Oracle WebCenter Content contains metadata used for accessing the file.

When a scheduled process has run to completion on a file, the record for the file includes a process ID.

**Accessing Content in a New Account**

When you create a new account in Oracle WebCenter Content and the content server is not restarted, access to content in the new account from the File Import and Export page may be delayed until the policy store is updated.
**Account Names**

If you create custom accounts for importing or exporting data, use the following conventions for naming the account: Do not include a slash "/" at the beginning or end End with "$" to avoid partial string matching Use "/" as a separator in the hierarchical structure.

For example: `fin$/journal$/import$` The File Import and Export page transforms account names by removing the $. For example `fin$/journal$/import$` displays as `fin/journal/import`. The Remote Introdoc Client (RIDC) HTTP command-line interface (CLI) transforms the account name you specify without $ symbols to one that includes them. For example, `fin/journal/import` becomes `fin$/journal$/import$` in WebCenter Content.

**Deleting Files**

You can delete one file at a time when you use the File Import and Export page. To delete multiple files simultaneously from the content repository, use the standard service page in Oracle WebCenter Content.

**External Data Integration Services for Oracle Cloud**

**External Data Integration Services for Oracle Cloud: Overview**

Use External Data Integration Services for Oracle Cloud to load data into Oracle Fusion Applications from external sources, such as legacy systems and third-party applications.

Components of External Data Integration Services for Oracle Cloud include:

- Templates and control files for formatting, structuring, and generating the data file.
- A general file load process for loading values from the data file into interface tables.
- Application-specific data import processes for transferring data from interface tables to the application tables in your Oracle Fusion Applications.

To use External Data Integration Services for Oracle Cloud to load data into Oracle Fusion Applications tables:

1. Prepare your data and generate a data file by using the product-specific templates and control files.
2. Transfer the data file to the integrated content management server.
3. Run the Load Interface File for Import process.
4. Correct data load errors, if necessary.
5. Run the appropriate application-specific process for validating and inserting the data into application tables.

6. Correct data import errors, if necessary.

For templates and control files, see assets with the File-Based Data Import type in Oracle Enterprise Repository for Oracle Fusion Applications (http://fusionappsor.oracle.com). For more information, see the Documentation tab for the Load Interface File for Import process in Oracle Enterprise Repository.

**Locating File Import Templates: Explained**

External data that you integrate into your Oracle Fusion Applications must be structured and formatted according to the properties of the fields and tables that store the data. To prepare external data so that data types, structural relationships, and other properties of the data correctly align to the data types, structural relationships, and properties of the target tables, use the product-specific templates and control files in Oracle Enterprise Repository for Oracle Fusion Applications.

You access these files from the Documentation tab of the scheduled process that corresponds to the interface tables that store the data. To find the process, you can search the interface table or you can search the specific process, if you know it.

Aspects of preparing external data using templates involve these tasks.

- Finding templates and control files
- Downloading templates
- Opening XLS templates

**Finding Templates and Control Files**

To find the templates and control files:

1. Sign in to Oracle Enterprise Repository.
2. Enter the following information in the Search fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search String</td>
<td>FBDI</td>
</tr>
<tr>
<td>Type</td>
<td>Scheduled Process</td>
</tr>
<tr>
<td>FusionApps: Logical Business Area</td>
<td>(Optional) Select the value relevant to your implementation.</td>
</tr>
</tbody>
</table>

3. Click Search.
4. Select Load Interface File for Import from the results.

**Downloading Templates**

To download the templates:
1. Use the Search area to locate the Load Interface File for Import job and then select it from the search results.

2. Click the Documentation tab in the lower pane to see a list of links to application-specific import jobs.

3. Click a link to access the job.

4. Click the Documentation tab in the lower pane to see a list of links that access:
   - Control files, which describe the logical flow of the data load process
   - XLS templates, which include worksheets and macros that assist you in structuring, formatting, and generating your data file

5. Click the link to download the file.

**Opening the XLS Template**

To prepare your data in a spreadsheet format, use XLS templates:

1. Open the XLS template.
   
   The first worksheet in each file provides instructions for using the template.

   **Important**
   
   If you omit or fail to complete the instructions, data load errors and data import failure are likely.

   2. Save a copy of the file.

   3. Click the **Generate CSV File** button.

   The macro generates a comma-separated values (CSV) file and compresses it into a ZIP file; you must transfer the ZIP file to the content management server.

**Using Excel Integration Templates to Generate Data Files: Points to Consider**

Oracle Enterprise Repository for Oracle Fusion Applications includes integration templates to help you prepare external data for loading and importing. Each template includes table-specific instructions, guidelines, formatted spreadsheets, and best practices for preparing the data file for upload. Use the templates to ensure that your data conforms to the structure and format of the target application tables.

**Templates**

This list details the characteristics of the templates:

- Each interface table is represented by a separate worksheet.

- Each interface table field is represented by a worksheet column with a header in the first row.
• Each column header contains bubble text, or comments, that include details about the column, such as the expected data type, length, and, in some cases, other instructional text.

• The worksheet columns appear in the order that the control file processes the data file.

• The columns that you do not intend to use can be hidden, but not reordered or deleted.

Important
Deleting or reordering columns will cause the load process to fail and result in an unsuccessful data load.

• The external data must conform to the data type that the control file and process for the associated database column accepts.

• Date column values must appear in the YYYY/MM/DD format.

• Amount column values must appear with no separators other than a period (.) as the decimal separator.

• Negative values must be preceded by the minus (-) sign.

• Column values that require whole numbers include data validation to allow whole numbers only.

• Columns are formatted, where applicable, to match the target field data type to eliminate data entry errors.

• For columns that require internal ID values, refer to the bubble text for additional guidance about finding these values.

• When using Microsoft Excel to generate or update the CSV file, you must select YYYY/MM/DD as your regional setting for date values.

Using XML Templates to Generate Data Files for Integration: Highlights

Oracle Enterprise Repository for Oracle Fusion Applications includes XML integration templates assets that you use with Oracle Data Integrator (ODI) to generate import files from your external data.

To use the XML templates and generate the import files, you must:

• Install and set up Oracle Data Integrator

• Create source and target models

• Create integration projects

Note
In Oracle Cloud implementations, you must upload the ZIP file to the content management repository in Oracle Cloud. In non-Cloud implementations, you
can streamline the data integration process by installing the content management document transfer utility so ODI performs the ZIP file transfer.

Oracle Data Integrator provides a solution for integrating complex data from a variety of sources into your Oracle Fusion applications. The Oracle Fusion Middleware Installation Guide for Oracle Data Integrator and the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator provide complete details pertaining to the installation and set up of this product.

Installing and Setting Up Oracle Data Integrator

- Install Oracle Data Integrator to use Oracle Fusion Applications XML integration templates. Refer to the Oracle Fusion Middleware Installation Guide for Oracle Data Integrator.

  See: Installing Oracle Data Integrator

- Set up Oracle Data Integrator to use Oracle Fusion Applications XML integration templates. Refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.

  See: Setting up the Topology

Creating Source and Target Models

- Create the ODI models for both the source and target datastores. You determine the source models that you use based on the system or technology of the external data that you want to import into your Oracle Fusion application. You create the target models by importing the XML files, which you download from Oracle Enterprise Repository. For more information, refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.

  See: Creating and Reverse-Engineering a Model

Configuring Integration Projects

- Create and configure an integration project, which entails selecting the knowledge modules, creating the interfaces, and mapping the source and target datastores. For more information, refer to the Oracle Fusion Middleware Developer’s Guide for Oracle Data Integrator.

  See: Creating an Integration Project

Using XML Integration Templates to Generate Data Files: Points to Consider

Use XML templates in Oracle Data Integrator to prepare your external data for loading and importing. Oracle Enterprise Repository for Oracle Fusion Applications includes three types of XML templates that you import as target models in your Oracle Data Integrator repository.

Oracle Enterprise Repository includes these three levels of XML files:

- Family-level
- Product-level
• Product

**Family-Level XML Files**

A family-level XML file is common to a group of product-level model folders and product models.

Consider the following points when you use family-level XML files:

- The family-level XML file supports all of the Oracle Enterprise Repository assets in the family, for example Oracle Fusion Financials or Human Capital Management.
- You import the family-level XML file into your Oracle Data Integrator repository prior to importing the other XML files.
- You import one family-level XML file as a model folder for each family of products.
- You import each family-level XML file as a top-level model folder.
- You import the family-level XML file one time; it supports all subsumed product-level model folders.
- You select Synonym mode Insert Update as the import type.

**Product-Level XML Files**

A product-level XML file is common to a group of product models.

Consider the following points when you use product-level XML files:

- The product-level XML file supports all of the Oracle Enterprise Repository assets in the product line, for example Fixed Assets, General Ledger, or Payables.
- You import one product-level XML file as a model folder for each line of products.
- You import the product-level XML file as a model folder into your Oracle Data Integrator repository after you import the family-level XML file, but before you import product XML files.
- You import each product-level XML file as a midlevel model folder within the appropriate family-level model folder.
- You import the product-level XML file one time; it supports all subsumed product models.
- You select Synonym mode Insert Update as the import type.

**Product XML Files**

A product XML file represents a specific Oracle Enterprise Repository interface table asset.

Consider the following points when you use product XML files:

- You import one product XML file as a model for each interface table or set of tables, for example Mass Additions.
• You import the product XML file as a model into your Oracle Data Integrator repository after you import the product-level XML file.
• You import each product XML file as a model within the appropriate product-level model folder.
• You import each product XML file one time.
• You select Synonym mode Insert Update as the import type.
• The model is based on File technology.
• After you import the product model, you connect the model to the correct logical schema.

Creating Integration Projects That Generate Data Files for Import: Points to Consider

When you use Oracle Data Integrator (ODI) to generate import data files from external data sources, you must configure an integration project. Integration projects are collections of ODI components that provide the procedural details of an integration from a source to a target. In this scenario, the source is your external data and the target is the import data file that you load and import into your Oracle Fusion Applications.

Configure these components to create your integration project.

• Knowledge modules
• Integration interfaces

Knowledge Modules

Knowledge modules contain the information that Oracle Data Integrator requires to perform a specific set of tasks against a specific technology or set of technologies. Examples include Check knowledge modules, which ensure that constraints on the sources and targets are not violated and integration knowledge modules, which load data to target tables.

Consider the following points about knowledge modules:

• The knowledge modules that you import into your integration project depend on the source and target technologies, as well as other integration-specific rules and processes.
• Multiple types of knowledge modules exist in ODI.
• Use the SQL File to Append module to create the import data file.

Integration Interfaces

Integration interfaces comprise sets of rules that define the loading of data from one or more sources to the target.

Consider the following points about integration interfaces:

• The source is the datastore from your external data model.
• The target is the interface table datastore, which is the CSV file from your interface table model.

• After you set up the source and target datastores, map the target fields to the source fields.

• You can map source field values to target fields or constants.

Transferring Data Files to Target Accounts in Oracle WebCenter Content: Explained

After you generate the ZIP file that contains the CSV data import file, transfer it to the content repository.

Use any of these methods to transfer file:

• File Import and Export page in Oracle Fusion Applications

• Oracle WebCenter Content Document Transfer Utility

• Oracle Fusion Financials Utility web service

Note

Consult Oracle Enterprise Repository for Oracle Fusion Applications for web service documentation.

Aspects of transferring data files to content management involve the following:

• Target accounts

• Accessing transferred content

Target Accounts

You must transfer files to these predefined account in content management that corresponds to the interface table or assets.

<table>
<thead>
<tr>
<th>Interface Table</th>
<th>Predefined Account</th>
</tr>
</thead>
<tbody>
<tr>
<td>Payables Standard Invoice Import</td>
<td>fin/payables/import</td>
</tr>
<tr>
<td>• AutoInvoice Import</td>
<td></td>
</tr>
<tr>
<td>• Receivables Standard Receipt Import</td>
<td>fin/receivables/import</td>
</tr>
<tr>
<td>• Customer Import</td>
<td></td>
</tr>
<tr>
<td>• China Value Added Tax Invoice Import</td>
<td></td>
</tr>
<tr>
<td>• BA12 Format Bank Statements Import</td>
<td>fin/cashManagement/import</td>
</tr>
<tr>
<td>• EDIFACT FINSTA Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>• ISO200022 CAMT053 Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>• SWIFT MT940 Format Bank Statements Import</td>
<td></td>
</tr>
<tr>
<td>Process</td>
<td>Import Path</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>Fixed Asset Mass Additions Import</td>
<td>fin/assets/import</td>
</tr>
<tr>
<td>Fixed Asset Mass Adjustments Import</td>
<td></td>
</tr>
<tr>
<td>Fixed Asset Mass Retirements Import</td>
<td></td>
</tr>
<tr>
<td>Fixed Asset Mass Transfers Import</td>
<td></td>
</tr>
<tr>
<td>Fixed Asset Units of Production Import</td>
<td></td>
</tr>
<tr>
<td>Intercompany Transaction Import</td>
<td>fin/intercompany/import</td>
</tr>
<tr>
<td>Journal Import</td>
<td>fin/generalLedger/import</td>
</tr>
<tr>
<td>Chart Of Accounts Segment Values and Hierarchies Import</td>
<td></td>
</tr>
<tr>
<td>General Ledger Budget Balance Import</td>
<td>fin/budgetBalance/import</td>
</tr>
<tr>
<td>Supplier Bank Account Import</td>
<td>fin/payables/import</td>
</tr>
<tr>
<td>Tax Configuration Content Import</td>
<td>fin/tax/import</td>
</tr>
<tr>
<td>Import Blanket Purchase Agreements</td>
<td>prc/blanketPurchaseAgreement/import</td>
</tr>
<tr>
<td>Import Contract Purchase Agreements</td>
<td>prc/contractPurchaseAgreement/import</td>
</tr>
<tr>
<td>Import Purchase Orders</td>
<td>prc/purchaseOrder/import</td>
</tr>
<tr>
<td>Import Requisitions</td>
<td>prc/requisition/import</td>
</tr>
<tr>
<td>• Import Suppliers</td>
<td>prc/supplier/import</td>
</tr>
<tr>
<td>• Import Supplier Sites</td>
<td></td>
</tr>
<tr>
<td>• Import Supplier Site Contacts</td>
<td></td>
</tr>
<tr>
<td>• Import Supplier Site Assignments</td>
<td></td>
</tr>
<tr>
<td>Project Enterprise Resource Import</td>
<td>prj/projectManagement/import</td>
</tr>
<tr>
<td>Project Unprocessed Expenditure Item Import</td>
<td>prj/projectCosting/import</td>
</tr>
<tr>
<td>Cycle Count Import</td>
<td>scm/cycleCount/import</td>
</tr>
<tr>
<td>Inventory Reservation Import</td>
<td>scm/inventoryReservation/import</td>
</tr>
<tr>
<td>Inventory Transaction Import</td>
<td>scm/inventoryTransaction/import</td>
</tr>
<tr>
<td>Item Import</td>
<td>scm/item/import</td>
</tr>
<tr>
<td>Receiving Receipt Import</td>
<td>scm/receivingReceipt/import</td>
</tr>
<tr>
<td>Shipment Request Import</td>
<td>scm/shipmentRequest/import</td>
</tr>
</tbody>
</table>

You can create subaccounts to further organize your files. However, you must create the account subordinate to the predefined account for the asset you are integrating.

**Accessing Transferred Content**

To access your transferred data you must access the account that corresponds to the interface table or asset appropriate for the data.

Available data integration processes move the content into and out of Oracle Fusion Applications tables. Running an import or export process creates a process ID in content management that you can use to identify the content you wish to overwrite or extract.
Oracle Enterprise Scheduler import process jobs result in the following hierarchy of items in Oracle WebCenter Content:

- A root import job is a list of all unprocessed files in an account. This job submits the child jobs that process each unprocessed file.
- A parent import job is a single file ID, account name, and the import steps (download, extract, import) for a single job, job set, or subrequests. This type of job tags the file with its request ID, provided the file is not deleted immediately after successful import.
- A child import job is a direct data load from a prepared file, typically a SQLLoader. Typically, the parent import job submits this job.

**Document Transfer Utility: Explained**

The WebCenter Content Document Transfer Utility for Oracle Fusion Applications is a feature-set Java library that provides programmatic access to the content repository. Use the utility to import and export documents, such as import files that contain external data that you want to load into interface and application tables.

The library includes:

- Oracle WebCenter Content client command line tool
- Oracle Data Integrator (ODI) upload and download tools
- Oracle WebCenter Content remote intradoc client (RIDC)
- Oracle HTTPClient
- Oracle Fusion Applications branding and defaults

Options for the WebCenter Content Document Transfer Utility for Oracle Fusion Applications fall into these categories:

- DownloadTool program options
- UploadTool program options
- Debugging and silent invocation options

**DownloadTool Program Options**

This table describes the download tool program options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>Protocol-specific connection URL of content server</td>
</tr>
<tr>
<td>username</td>
<td>Username to leverage</td>
</tr>
<tr>
<td>password</td>
<td>Password, supplied in command line</td>
</tr>
<tr>
<td>passwordFile</td>
<td>Password, supplied in text file on the first line of the file</td>
</tr>
</tbody>
</table>
### External Integration

#### Download Tool

- **dID**: ID of document revision to download
  - dID is unique across repository
  - dID changes with each revision
  - **Note**: Alternatively, specify the `dDocName` and `RevisionSelectionMethod` to identify the dID to leverage.

- **dDocName**: Content name
  - Multiple revisions of a document can share the same `dDocName` value, otherwise it is unique.
  - **Note**: You should also provide the `RevisionSelectionMethod` value.

- **RevisionSelectionMethod**: Revision to download
  - Valid values: Latest, LatestReleased
  - Default value: Latest

- **outputFile**: Path and name of local file to write

**Here you see a sample download invocation command:**

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg
username=weblogic
password=we1com3i
dID=21537
outputFile="/tmp/output.doc"
```

**Here you see sample output:**

Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing download (GET_FILE) ...
Download successful.

### Upload Tool Program Options

This table describes the upload tool program options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>Protocol-specific connection URL of content server</td>
</tr>
<tr>
<td>username</td>
<td>Username to leverage</td>
</tr>
<tr>
<td>password</td>
<td>Password, supplied in command-line</td>
</tr>
<tr>
<td>passwordFile</td>
<td>Password, supplied in text file on the first line of the file</td>
</tr>
<tr>
<td>primaryFile</td>
<td>Fully-qualified path of local primary file to upload</td>
</tr>
<tr>
<td>dDocAccount</td>
<td>Destination account</td>
</tr>
</tbody>
</table>
Here you see a sample upload invocation command:

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
  oracle.ucm.client.UploadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic
  password=welcome3
primaryFile="/tmp/resume.doc" dDocTitle="Resume of MSMITH" -dDocAccount=/
  acme/sales
```

Here you see sample output:

```
Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing upload (CHECKIN_UNIVERSAL) ...
Upload successful.
[dID=21537 | dDocName=UCMFA021487]
```

### Debugging and Silent Invocation Options

This table describes the options which are common to all tools that the invoker can leverage.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>verbose</strong></td>
<td>Verbose output</td>
</tr>
<tr>
<td></td>
<td>Log filled with Request/Response DataBinders</td>
</tr>
<tr>
<td><strong>quiet</strong></td>
<td>Minimal output</td>
</tr>
<tr>
<td><strong>version</strong></td>
<td>Print tool revision or version</td>
</tr>
<tr>
<td><strong>log_file_name</strong></td>
<td>Send program output to specified log file instead of the System.out log file</td>
</tr>
<tr>
<td><strong>log_file_append</strong></td>
<td>Append log to existing log file rather than overwrite it</td>
</tr>
<tr>
<td></td>
<td>Valid values: true, false</td>
</tr>
<tr>
<td></td>
<td>Default value: false</td>
</tr>
<tr>
<td><strong>socketTimeout</strong></td>
<td>Override time out of socket</td>
</tr>
<tr>
<td></td>
<td>Specify override time in seconds</td>
</tr>
</tbody>
</table>

You can use the tools to test the connection. Provide only the url, username, and password as you see in this sample test:
java -classpath "oracle.ucm.fa_client_11.1.1.jar"
oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic
password=we1com3!

Here you see the sample output:

Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing connection test (PING_SERVER) ...
Connection test successful.

Load Interface File for Import Process

Use to load external setup or transaction data from a data file in the content repository to interface tables. The process prepares the data for import into application tables.

You run this process from the Scheduled Processes page. You can run it on a recurring basis.

Before running this process, you must:

1. Prepare your data file.
2. Transfer the data file to the content repository.

Parameters

Import Process

Select the target import process.

Data file

Enter the relative path and the file name of the *.zip data file in the content repository.

Importing Data into Application Tables: Procedure

The final destination for your external data is the application data tables of your Oracle Fusion Applications product.

Aspects of importing data into application tables involve the following:

• Loading data into interface tables
• Finding and submitting the import process

Loading Data into Interface Tables

Interface tables are intermediary tables that store your data temporarily while the system validates format and structure. Run the Load Interface File for Import
scheduled process to load data from the data file into the interface table that corresponds to the template that you use to prepare the data.

To load your data into interface tables, submit the Load Interface File for Import scheduled process:

1. Sign in to Oracle Fusion Applications.
2. In the Navigator menu, select Tools, Scheduled Processes
3. Click the Schedule New Process button.
4. Search and select the Load Interface File for Import job.
5. When the Process Details page appears:
   a. Select the target import process.
   b. Enter the data file name.

**Note**

If the file exists in an account subordinate to the predefined account, you must enter the entire path relative to the predefined account in the content repository. Include all subaccounts and the file name.

6. Submit the process.

   If no errors exist in the data file, then the process populates the interface tables.

**Note**

The data file remains in the content repository after the process ends.

**Finding and Submitting the Import Process**

Run the appropriate import process to import the data into the interface tables of your Oracle Fusion Applications product.

To import your data:

1. Sign in to Oracle Fusion Applications.
2. In the Navigator menu, select Tools, Scheduled Processes
3. Click the Schedule New Process button.
4. Find and select the import process that is specific to the target application tables.
5. When the Process Details page appears, select the process that corresponds to the data that you are importing.

   If you prepared your data using the spreadsheet template, select the process named in the Overview section of the spreadsheet.

6. Submit the process.

**Note**
For more detailed information on the process used for data prepared using the spreadsheet template, see the Instructions and CSV Generation tab of the spreadsheet template.

**Correcting Import Load Process Errors: Explained**

The Load Interface File for Import process ends in error if the load of the data file fails on any row.

The following conditions apply when the process ends in error:

- The Load File to Interface child process ends in either warning or error.
- All rows that were loaded by the process are deleted, even those rows that loaded successfully.

To correct errors:

1. Review the error logs.
2. Change any formatting or structural anomalies that exist in the data.
3. Recreate the CSV and ZIP files.
4. Transfer the file to the content management server.
5. Submit the Load Interface File for Import job.
6. Repeat these steps until the process successfully loads the data.
7. Import the data using the appropriate product-specific process.
Importing and Exporting Setup Data

Configuration Packages: Explained

Almost all Oracle Fusion application implementations require moving functional setup data from one instance into another at various points in the lifecycle of the applications. For example, one of the typical cases in any enterprise application implementation is to first implement in a development or test application instance and then deploy to a production application instance after thorough testing. You can move functional setup configurations of applications from one application instance into another by exporting and importing Configuration packages from the Manage Configuration Packages page.

A Configuration Package contains the setup import and export definition. The setup import and export definition is the list of setup tasks and their associated business objects that identifies the setup data for export as well as the data itself. When you create a configuration package only the setup export and import definition exists. Once you export the configuration package appropriate setup data is added to the configuration package using the definition. Once a configuration package is exported, the setup export and import definition is locked and cannot be changed.

You generate the setup export and import definition by selecting an implementation project and creating a configuration package. The tasks and their associated business objects in the selected implementation project define the setup export and import definition for the configuration package. In addition, the sequence of the tasks in the implementation project determine the export and import sequence.

Exporting and Importing Setup Data: Explained

A configuration package is required to export setup data. You can export a configuration package once you create it, or at any time in the future. During export, appropriate setup data will be identified based on the setup export definition and added to the configuration package. The setup data in the configuration package is a snapshot of the data in the source application instance at the time of export. After the export completes, you can download the configuration package as a zipped archive of multiple XML files, move it to the target application instance, and upload and import it.
Export
You can export a configuration package multiple times by creating multiple versions. While the export definition remains the same in each version, the setup data can be different if you modified the data in the time period between the different runs of the export process. Since each version of the configuration package has a snapshot of the data in the source instance, you can compare and analyze various versions of the configuration package to see how the setup data changed.

Import
In the target application instance, the setup import process will insert all new data from the source configuration package that does not already exist and update any existing data with changes from the source. Setup data that exists in the target instance but not in source will remain unchanged.

Export and Import Reports
You can review the results of the export and import processes using reports. The results appear ordered by business objects and include information on any errors encountered during the export or import process. If a setup export or import process paused due to errors encountered or for a manual task to be performed outside of the application, then you can resume the paused process.

These reports show what setup data was exported or imported and by which specific process. You can change the reports to validate the setup data as well as to compare or analyze it. A report is generated for each business object. These reports show the same information as the export and import results seen directly in the application.

Process status details are available as text files showing the status of an export or import process including the errors encountered during the process.

Moving Common Reference Objects

Moving Common Reference Objects: Overview
The common reference objects in Oracle Middleware Extensions for Applications are used by several setup tasks in the Setup and Maintenance work area. The common reference objects become a part of the configuration package that is created for an implementation project. While moving the application content, for example, from the test phase to the production phase of an implementation, you must pay special attention to the nuances of these common reference objects.

Parameters
The common reference objects are represented as business objects. A single object can be referenced in multiple setup tasks with different parameters. In the configuration package that is created for the implementation project, parameters passed to a setup task are also passed to the business objects being moved. As a result, the scope of the setup tasks is maintained intact during the movement.
**Dependencies**

Common reference objects may have internal references or dependencies among other common reference objects. Therefore, it is necessary that all the dependencies are noted before the movement of objects so that there are no broken references among the objects.

**Business Objects for Moving Common Reference Objects: Points to Consider**

Common reference objects in Oracle Fusion Functional Setup Manager are represented by business objects. These business objects are the agents that contain the application content and carry them across whenever the application setup is moved from one environment to another, for example, test environment to production environment.

**Choice of Parameters**

The following table lists the business objects, the corresponding movement details, and the effect of the setup task parameter on the scope of the movement.

**Note**

- Only the translation in the current user language is moved.
- The Oracle Social Network business objects and the Navigator menu customizations are moved using the customization sets on the Customization Migration page instead of using the export and import function in the Setup and Maintenance work area.

<table>
<thead>
<tr>
<th>Business Object Name</th>
<th>Moved Functional Item</th>
<th>Effect on the Scope of Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Message</td>
<td>Messages and associated tokens</td>
<td>No parameters: all messages are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleName/applicationId: only the specified message is moved.</td>
</tr>
<tr>
<td>Application Taxonomy</td>
<td>Application taxonomy modules and components</td>
<td>No parameters: all taxonomy modules and components are moved.</td>
</tr>
<tr>
<td>Application Attachment Entity</td>
<td>Attachment entities</td>
<td>No parameters: all attachment entities are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only attachment entities belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td>Application Attachment Category</td>
<td>Attachment categories and category-to-entity mappings</td>
<td>No parameters: all attachment categories and category-to-entity mappings are moved. 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. moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. code/applicationId: only the specified document sequence category code is moved.</td>
</tr>
<tr>
<td>Application Document Sequence</td>
<td>Document sequences and their assignments</td>
<td>No parameters: all sequences are moved. moduleType/moduleKey: only document sequences belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. name: only the specified document sequence is moved.</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. |
|----------------------------------|-----------------------------------------------|--------------------------------------------------------------------------------|

**Note**
Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.

**Note**
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.

| Application Extensible Flexfield | Extensible flexfield registration data and setup data, including categories | No parameters: all extensible flexfields are moved.  
moduleType/moduleKey: only extensible flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
extensibleFlexfieldCode/applicationId: only the specified extensible flexfield is moved. |
|---------------------------------|---------------------------------------------------------------|--------------------------------------------------------------------------------|

**Note**
Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.

**Note**
Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.
| Application Key Flexfield | Key flexfield registration data and setup data | No parameters: all key flexfields are moved.  
*moduleType/moduleKey*: only key flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
*keyFlexfieldCode/applicationId*: only the specified key flexfield is moved.  

**Note**  
Importing a flexfield’s metadata can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.

| Application Flexfield Value Set | Value set setup data | No parameters: all value sets are moved.  
*moduleType/moduleKey*: only value sets belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  
*valueSetCode*: only the specified value set is moved.  

**Note**  
Importing a value set’s metadata can change the deployment status of flexfields that use the value set, and therefore the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.

| Application Reference Currency | Currency data | No parameters: all currencies are moved.  
| Application Reference ISO Language | ISO language data | No parameters: all ISO languages are moved.  
| Application Reference Industry | Industry data including industries in territories data | No parameters: all industries are moved.  

13-6 Oracle Project Portfolio Management Cloud Implementing Project Execution Management
<table>
<thead>
<tr>
<th>Application Reference Language</th>
<th>Language data</th>
<th>No parameters: all languages are moved.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Reference Natural Language</td>
<td>Natural language data</td>
<td>No parameters: all natural languages are moved.</td>
</tr>
<tr>
<td>Application Reference Territory</td>
<td>Territory data</td>
<td>No parameters: all territories are moved.</td>
</tr>
<tr>
<td>Application Reference Time zone</td>
<td>Time zone data</td>
<td>No parameters: all time zones are moved.</td>
</tr>
<tr>
<td>Application Standard Lookup</td>
<td>Standard lookup types and their lookup codes</td>
<td>No parameters: all standard lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Common Lookup</td>
<td>Common lookup types and their lookup codes</td>
<td>No parameters: all common lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified common lookup is moved.</td>
</tr>
<tr>
<td>Application Set-Enabled Lookup</td>
<td>Set-enabled lookup types and their lookup codes</td>
<td>No parameters: all set-enabled lookups are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>lookupType: only the specified set-enabled lookup is moved.</td>
</tr>
<tr>
<td>Application Profile Category</td>
<td>Profile categories</td>
<td>No parameters: all profile categories are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>moduleType/moduleKey: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>name/applicationId: only the specified category is moved.</td>
</tr>
</tbody>
</table>
| Application Profile Option          | Profile options and their values                                                                 | No parameters: all profile options and their values are moved.  
|                                   |                                                                                                   | moduleType/moduleKey: only profile options and their values belonging to the specified module are moved.  
|                                   |                                                                                                   | profileOptionName: only the specified profile option and its values are moved.  
| Application Profile Value         | Profile options and their values                                                                 | No parameters: all profiles and their values are moved.  
|                                   |                                                                                                   | moduleType/moduleKey: only profiles and their values belonging to the specified module are moved.  
|                                   |                                                                                                   | categoryName/categoryApplicationId: only profiles and their values belonging to the specified category are moved.  
|                                   |                                                                                                   | profileOptionName: only the specified profile and its values are moved.  
| Application Reference Data Set    | Reference data sets                                                                                | No parameters: all sets are moved.  
| Application Reference Data Set    | Reference data set assignments                                                                    | determinantType: only assignments for the specified determinant type are moved.  
| Assignment                        |                                                                                                   | determinantType/referenceGroupName: only assignments for the specified determinant type and reference group are moved.  
| Application Tree Structure        | Tree structures and any labels assigned to the tree structure                                     | No parameters: all tree structures (and their labels) are moved.  
|                                   |                                                                                                   | moduleType/moduleKey: only tree structures (and their labels) belonging to the specified module are moved.  
|                                   |                                                                                                   | treeStructureCode: only the specified tree structure (with its labels) is moved.  

<table>
<thead>
<tr>
<th>Application Tree</th>
<th>Tree codes and versions</th>
<th>No parameters: all trees are moved.</th>
</tr>
</thead>
<tbody>
<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>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>Note</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.

action
The kind of access named in a security policy, such as view or edit.

ADF
Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

application identity
Predefined application level user with elevated privileges. An application identity authorizes jobs and transactions for which other users are not authorized, such as a payroll run authorized to access a taxpayer ID while the user who initiated the job is not authorized to access such personally identifiable information.

application 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.

available hours
Hours on the resource's calendar that have been, or can be, consumed with project assignments and nonproject events.

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.

chart of accounts

The account structure your organization uses to record transactions and maintain account balances.

competency

A type of qualification that represents a piece of knowledge, a skill, an aptitude, or an attribute that is measurable and demonstrated by a resource in the work context.

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.

**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.

**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.

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.

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.

flexfield
Grouping of extensible data fields called segments, where each segment is an attribute added to an entity for capturing additional information.

flexfield segment
An extensible data field that represents an attribute on an entity and captures a single atomic value corresponding to a predefined, single extension column in the Oracle Fusion Applications database. A segment appears globally or based on a context of other captured information.

function security
The control of access to a page or a specific widget or functionality within a page. Function security controls what a user can do.

global area
The region across the top of the user interface. It provides access to features and tools that are relevant to any page you are on.

HCM
Abbreviation for Human Capital Management.

HCM data role
A job role, such as benefits administrator, associated with specified instances of Oracle Fusion Human Capital Management (HCM) data, such as one or more positions or all persons in a department.

HCM person
A named person in the HCM database with a person type of Employee or Contingent Worker and an active assignment. In Oracle Fusion Project Resource Management, if you associate an HCM person with a project enterprise labor
resource, you can enable the resource's eligibility to fulfill project resource requests.

**identity**
A person representing a worker, supplier, or customer.

**import**
In the context of data integration, the transfer of data from interface tables to *application tables*, where the data is available to application users.

**inventory organization**
An organization that tracks inventory transactions and balances, and can manufacture or distribute products.

**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 structure**
The arrangement of segments in a key flexfield. In some cases, multiple structures can be defined for a single key flexfield.

**keyword**
A word or phrase, entered as free-form, unstructured text on a project resource request, that does not exist as a predefined qualification content item. Keywords are matched against the resource’s qualifications and the results are included in the qualification score calculation.

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

**legal employer**
A legal entity that employs people.
legal entity
An entity is identified and given rights and responsibilities under commercial law, through the registration with the country’s appropriate authority.

legal jurisdiction
A physical territory, such as a group of countries, single country, state, county, parish, or city, which comes under the purview of a legal authority.

legal reporting unit
The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.

legislative data group
A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.

load
In the context of data integration, the transfer of external data from data files to the receiving interface tables in preparation for an import into application tables.

lookup code
A value available for lookup within a lookup type such as the code BLUE within the lookup type COLORS.

lookup type
A set of lookup codes to be used together as a list of values on a field in the user interface.

mainline
A branch of data that serves as a single source of truth.

managed project enterprise labor resource
A project enterprise labor resource who is associated with a named person in HCM, and whose availability and staffing are managed in Oracle Fusion Project Resource Management. You use managed project enterprise labor resources to fulfill project resource requests.

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.

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.

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 calendar
Defines the work schedule for project assignments. The project calendar includes working days and hours, such as Monday through Friday for 8 hours a day, and any exceptions, such as holidays.

project enterprise labor resource
A labor resource that you can assign to multiple projects.

project expenditure organization
An organization that can incur expenditures and hold financial plans for projects.

project resource request
List of criteria used to find a qualified resource to fulfill an open resource demand on a project. Project resource requests include qualifications, keywords, requested date range, and other assignment information, such as project role and work location.

project unit
An operational subset of an enterprise, such as a line of business, that conducts business operations using projects, and needs to enforce consistent project planning, management, analysis, and reporting.
**projected utilization**
Percentage of hours that a resource or resources are assigned to work on project assignments compared to the available hours.

**qualification**
Items in structured content types such as competencies, degrees, and language skills that have specific values and proficiency ratings.

**qualification score**
A comparison of a resource's qualifications to the requested qualifications and keywords on a project resource request, expressed as a percentage.

**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.

**registration**
The record of a party’s identity related details with the appropriate government or legal authorities for the purpose of claiming and ensuring legal and or commercial rights and responsibilities.

**resource**
People designated as able to be assigned to work objects, for example, service agents, sales managers, or partner contacts. A sales manager and partner contact can be assigned to work on a lead or opportunity. A service agent can be assigned to a service request.

**resource breakdown structure**
One or more hierarchies of resources, resource types, resource formats, or other resource groupings that are used for financial and project planning and for viewing planned and actual amounts for a project.
resource calendar
A schedule of a resource's available work days and hours, such as Monday through Friday, 8 hours a day.

resource pool
A logical group of resources organized in a hierarchy for purposes of staffing, management, and reporting on utilization.

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.

segment
See
segregation of duties
An internal control to prevent a single individual from performing two or more phases of a business transaction or operation that could result in fraud.

set 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.

SQL predicate
A type of condition using SQL to constrain the data secured by a data security policy.

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.

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.