

**Oracle® Fusion Applications Customer Data Management
Implementation Guide**

11g Release 7 (11.1.7)

Part Number E20433-07

March 2013

Oracle® Fusion Applications Customer Data Management Implementation Guide

Part Number E20433-07

Copyright © 2011-2013, Oracle and/or its affiliates. All rights reserved.

Author: Abhishek Sura

Contributors: Brian Casey, Rakesh Chaudhary, Peggy Larson, Vijay Tiwary, Jacqueline Wood, Asra Alim, Catherine Bauer, Tina Brand, Angela Brown, Hema Hardikar, Essan Ni Jirman, Mary Kalway, Suzanne Kinkead, Mike Lavery, Kristin Penaskovic, P. S. G. V. Sekhar, Barbara Snyder, Rick Stober, Srinivas Vellikad, Megan Wallace, Jiri Weiss, Kathryn Wohnoutka

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

1 Overview

Customer Data Management Offering: Overview	1-1
Getting Started with an Implementation: Overview	1-2
Manage Application Implementation	1-3

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

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

3 Common Applications Configuration: Define Implementation Users

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

4 Common Applications Configuration: Define Currencies and Currency Rates

Manage Currencies	4-1
Manage Conversion Rate Types	4-3

5 Common Applications Configuration: Define Enterprise Structures for Customer Data Management

Enterprise Structures: Overview	5-1
Enterprise Structures Business Process Model: Explained	5-3
Global Enterprise Configuration: Points to Consider	5-5
Modeling Your Enterprise Management Structure in Oracle Fusion: Example	5-6
Essbase Character and Word Limitations	5-9
Define Initial Configuration with the Enterprise Structures Configurator	5-11
Define Reference Data Sharing	5-36
Define Enterprise: Manage Enterprise HCM Information	5-40
Define Enterprise: Manage Locations	5-41
Define Geographies	5-43
Define Legal Entities for Customer Data Management: Manage Legal Entity	5-64
Define Legal Entities for Customer Data Management: Manage Legal Entity HCM Information	5-69
Define Business Units for Customer Data Management: Manage Service Provider Relationships	5-74
Define Business Units for Customer Data Management: Specify Customer Contract Management Business Function Properties	5-77

Define Business Units for Customer Data Management: Specify Supplier Contract Management Business Function Properties	5-80
Define Business Units for Customer Data Management: Assign Business Unit Business Function	5-82
Define Business Units for Customer Data Management: Manage Business Units	5-84
Define Workforce Structures for CRM: Manage Locations	5-86
Define Workforce Structures for CRM: Manage Divisions	5-87
Define Workforce Structures for CRM: Manage Departments	5-90
Define Workforce Structures for CRM: FAQs for Manage Job Families	5-93
Define Workforce Structures for CRM: Manage Jobs	5-94
Jobs: Example	5-95

6 Common Applications Configuration: Define Security for CRM

Security Tasks: Highlights	6-1
Defining Security After Enterprise Setup: Points to Consider	6-4
Security Tasks and Oracle Fusion Applications: How They Fit Together	6-8
Security Tasks: Overview	6-11
Define Data Security	6-15
Define Users	6-37

7 Common Applications Configuration: Define Automated Governance, Risk, and Performance Controls

Segregation of Duties: Explained	7-1
Segregation of Duties in the Security Reference Implementation: Explained	7-2
Defining Segregation of Duties Policies: Points To Consider	7-4
Managing Segregation of Duties Risks and Violations: Critical Choices	7-6
Role Provisioning and Segregation of Duties: How They Work Together	7-8

8 Common Applications Configuration: Define Approval Management for Customer Data Management

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

9 Common Applications Configuration: Define Help Configuration

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

10 Common Applications Configuration: Define Application Toolkit Configuration

Define Application Toolkit Configuration: Overview	10-1
--	------

Map Reports to Work Areas	10-1
Set Watchlist Options	10-3

11 Common Applications Configuration: Maintain Common Reference Objects

Maintain Common Reference Objects: Overview	11-1
Define Application Taxonomy	11-1
Define Reference Data Sharing	11-4
Define Lookups	11-7
Manage Messages	11-14
Define ISO Reference Data	11-16
Manage Data Security Policies	11-19
Define Document Sequences	11-29
Manage Public Unique Identifier Profile Options	11-32
Define Trees	11-34
Define Profile Options	11-49
Define Flexfields	11-58
Define Attachments	11-102
Set Activity Stream Options	11-106
Manage Menu Customizations	11-106
Manage Audit Policies	11-107
Manage Oracle Social Network Objects	11-109
Manage Applications Core Common Reference Objects: Manage Applications Core Messages	11-113
Manage Applications Core Common Reference Objects: Manage Applications Core Administrator Profile Values	11-113

12 Common Applications Configuration: Define WebLogic Communication Services Configuration

Click-to-Dial: Explained	12-1
Click-to-Dial: Top Tasks	12-4
Configuring PSTN Gateway Address Using Topology Manager: Worked Example	12-5

13 Common Customer Data Management Configuration: Define Source Systems

Source Systems: Explained	13-1
Source System Entities: Explained	13-2
FAQs for Define Source Systems	13-2

14 Common Customer Data Management Configuration: Define Party Usages

Party Usage: Explained	14-1
Party Usage Filter Rules Explained	14-2
FAQs for Define Party Usages	14-2

15 Common Customer Data Management Configuration: Define Names

Additional Name Types: Explained	15-1
16 Common Customer Data Management Configuration: Define Identifiers	
Identifier Types: Explained	16-1
17 Common Customer Data Management Configuration: Define Relationships	
Relationships Types: Explained	17-1
Relationship Groups: Explained	17-2
Manage Relationship Types	17-2
Manage Relationship Groups	17-4
18 Common Customer Data Management Configuration: Define Classifications	
Classifications: Explained	18-1
FAQs for Define Classifications	18-2
19 Common Customer Data Management Configuration: Define Data Formats	
Data Formats and Setup Activities : How They Fit Together	19-1
Setup Formats: Explained	19-2
Creating an Address Style Format: Worked Example	19-2
Creating a Name Style Format: Worked Example	19-4
FAQs for Define Data Formats	19-6
20 Common Customer Data Management Configuration: Define Contact Points	
Telephone Numbering Plan: Explained	20-1
Updating Telephone Numbering Plans: Explained	20-1
How Setup Telephone Numbering Plans Fits in with the Other Setup Tasks	20-2
FAQs for Define Contact Points	20-3
21 Common Customer Data Management Configuration: Define Trading Community Import	
Trading Community Model Data Import Objects: Explained	21-1
Using Data Quality Services During Data Import: Explained	21-1
Defining the Import Process for Customers and Consumers: Points to Consider	21-2
Defining the Import Process for Customers and Consumers: Worked Example	21-4
Defining the Import Process for Resources and Partners: Points to Consider	21-7
Defining the Import Process for Resources and Partners: Worked Example	21-8
Performing What-If Analysis on Data Import Batches: Worked Example	21-9
FAQs for Define Trading Community Import	21-11
22 Common Customer Data Management Configuration: Define Resource Information	

Resource Directory: Explained	22-1
Setting up Resources: Explained	22-1
Identifying Resources: Explained	22-2
Resource Skills and Resource Assignment: Explained	22-2
Resources: How They Work within a Team	22-2
FAQs for Define Resource Information	22-2

23 Common Customer Data Management Configuration: Define Resource Role Information

Define Resource Roles: Explained	23-1
Provisioning Security Roles: Explained	23-1
Job-to-Role Mapping: Explained	23-2
Resource Role Assignment: Explained	23-2
Resource Team Membership and Role Assignment Components: How They Work Together	23-2
FAQs for Define Resource Role Information	23-3

24 Common Customer Data Management Configuration: Define Resource Organization Information

Resource Organizations and Organization Usage: Explained	24-1
Primary Resource Organization by Usage Explained	24-1
Resource Organization Hierarchy Versioning: Explained	24-1
Organization Memberships and Role Assignments: Explained	24-2
Create an Organization with Sales Usage: Worked Example	24-2
Adding a Sales Organization to a Sales Organization Hierarchy: Worked Example	24-3
Assigning a Resource to an Organization: Example	24-4
FAQs for Define Resource Organization Information	24-6

25 Common Customer Data Management Configuration: Define Resource Team Information

Resource Setup and Resource Role Setup: How They Fit Together	25-1
FAQs for Define Resource Team Information	25-1
What's a duty role	25-2

26 Common Customer Data Management Configuration: Manage Calendar Profile Option

Creating the CRM Common Calendar: Worked Example	26-1
--	------

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

Define File-Based Data Import	27-1
Getting Started with File-Based Import: Documentation Overview	27-27
Importing Consumers	27-28
Importing Contacts	27-36
Importing Country Structures	27-47
Importing Customer Hierarchies	27-51

Importing Customers	27-54
Importing Employee Resources	27-65
Importing Geographies	27-70
Importing Group Customers	27-73
Importing Legal Entities	27-81
Importing Resource Teams	27-92
FAQs for Define File-Based Data Import	27-96

28 Common Customer Data Management Configuration: Manage Bulk Data Export

Bulk Export: Overview	28-1
FAQs for Manage Bulk Data Export	28-5

29 Define Data Quality

Define Data Quality: Explained	29-1
Manage Data Quality Engine	29-3
Manage Server Configurations	29-23
Manage Data Quality Engine Matching Setup	29-25
Manage Matching Configurations	29-29
Manage Data Quality Engine Cleansing Setup	29-39
Manage Cleansing Configurations	29-40
Manage Data Synchronization	29-44
Troubleshooting Oracle Fusion Data Quality Services and IIR Servers	29-47
FAQs for Define Data Quality	29-65

30 Define Customer Hub Configuration

Run Request Dispatch Job	30-1
Manage Customer Hub Profile Options	30-1
Manage Survivorship Rules	30-3
Manage Party Center Trees	30-6
Manage Customer Data Hub Notes	30-9
Manage Agreement Rules	30-10
Enter Merge Request	30-13

31 Define Extensions for Customer Data Management

Define Custom Enterprise Scheduler Jobs	31-1
Customization and Sandboxes	31-3
FAQs for Define Extensions for Customer Data Management	31-19

32 Importing and Exporting Setup Data

Configuration Packages: Explained	32-1
Exporting and Importing Setup Data: Explained	32-1
Moving Common Reference Objects	32-2

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.



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. You can access the guides from the **Guides** menu in the global area at the top of Oracle Fusion Applications Help pages.

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.

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

For guides that are not available from the Guides menu, 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.

Note

The content of Oracle Enterprise Repository reflects the latest release of Oracle Fusion Applications.

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 the **Send Feedback to Oracle** link in the footer of Oracle Fusion Applications Help.

Overview

Customer Data Management Offering: Overview

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

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

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

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

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

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

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

Define Common Applications Configuration for Customer Data Management

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

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

Define Common Customer Data Management Configuration

Define the configuration for common setup such as users, enterprise structures, security, approval management, and common reference objects for customer data management.

Define Data Quality

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

Define Customer Hub Configuration

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

Define Extensions for Customer Data Management

Define extensions such as custom Oracle Enterprise Scheduler jobs.

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

Getting Started with an Implementation: Overview

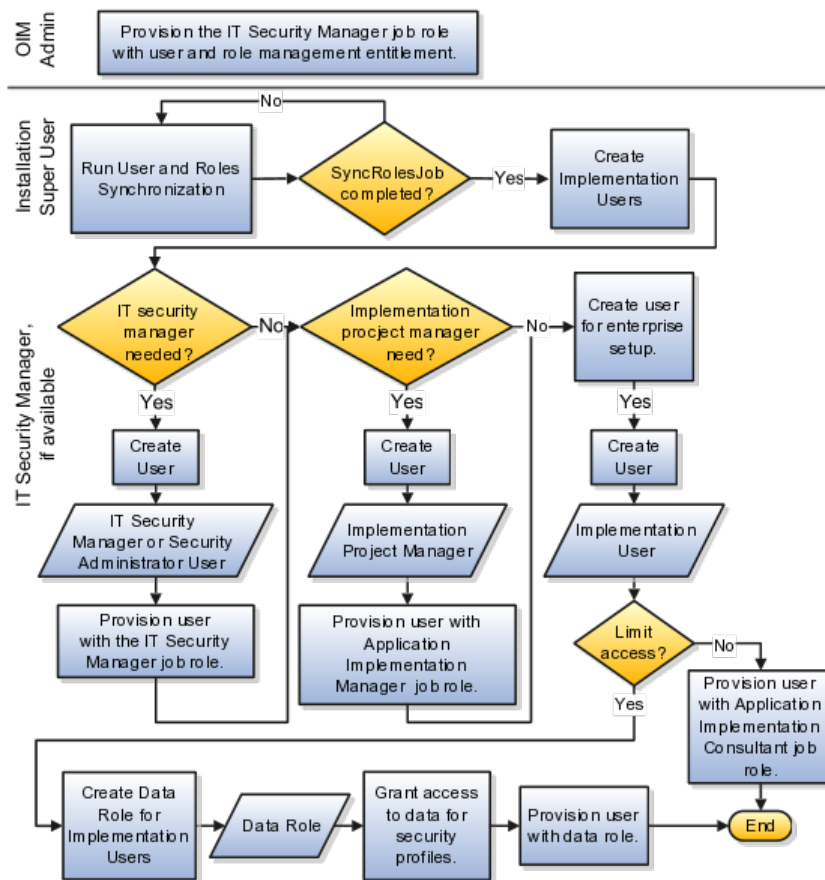
To start an Oracle Fusion Applications implementation, you must set up one or more initial users using the super user that was created during installation and provisioning of the Oracle Fusion Applications environment, or using the initial administrator user provided by Oracle for Oracle Cloud Application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

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

1. If you are not starting an Oracle Cloud Application Services implementation, sign into Oracle Identity Manager (OIM) as the OIM Administration users and provision the IT Security Manager job role with roles for user and role management. This enables the super user account, which is provisioned with the IT Security Manager job role, to create implementation users.
2. For starting all implementations, sign in as the user with initial access: either the Oracle Fusion Applications installation super user or the initial Oracle Cloud Application Services administrator user.
3. Select an offering to implement, and generate the setup tasks needed to implement the offering.
4. Perform the following security tasks:
 - a. Synchronize users and roles in the Lightweight Directory Access Protocol (LDAP) store with HCM user management by using the Run User and Roles Synchronization Process task.
 - 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 object.
 - 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 options. You create a project by selecting the offerings and options you want to implement together. You manage the project as a unit throughout the implementation lifecycle. You can assign these tasks to users and track their completion using the included project management tools.

Maintaining Setup Data

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

Exporting and Importing

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

Selecting Offerings

When creating an implementation project you see the list of offerings and options that are configured for implementation. Implementation managers specify which of those offerings and options to include in an implementation project. There are no hard and fast rules for how many offerings you should include in one implementation project. The implementation manager should decide based on how they plan to manage their implementations. For example,

if you will implement and deploy different offerings at different times, then having separate implementation projects will make it easier to manage the implementation life cycles. Furthermore, the more offerings you included in an implementation project, the bigger the generated task list will be. This is because the implementation task list includes all setup tasks needed to implement all included offerings. Alternatively, segmenting into multiple implementation projects makes the process easier to manage.

Offerings: Explained

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

Implementation Task Lists

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

Enabling Offerings

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

Provisioning Offerings

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

Options: Explained

Each offering in general includes a set of standard functionality and a set of optional modules, which are called options. For example, in addition to standard Opportunity Management, the Sales offering includes optional functionality such as Sales Catalog, Sales Forecasting, Sales Prediction Engine, and Outlook Integration. These optional functions may not be relevant to all application

implementations. Because these are subprocesses within an offering, you do not always implement options that are not core to the standard transactions of the offering.

Feature Choices: Explained

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

You can configure feature choices in three different ways:

Yes or No

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

Single Select

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

Multi-Select

If the feature has multiple choices but one or more can be applicable to an implementation then all choices are presented with a checkbox. Select all that apply by checking the appropriate choices.

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

User and Role Synchronization: Explained

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

Note

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

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

Tip

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

Common Applications Configuration: Define Implementation Users

Initial Security Administration: Critical Choices

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

Initial security administration consists of the following.

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

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

Preparing the IT Security Manager Job Role

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

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

- Identity User Administrators
- Role Administrators

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

Synchronizing Users and Roles from LDAP

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

Defining Initial Implementation Users

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

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

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

Creating Implementation Users

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

Note

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

Creating Data Roles for Implementation Users

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

Provisioning Roles to Implementation Users

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

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

Caution

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

Initial Security Administration: Worked Example

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

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

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

The following table summarizes key decisions for this scenario.

Decision	In this Example
How to sign in to Oracle Fusion Applications for the first time	Use the super user account that was created when installing and provisioning Oracle Fusion Applications (for example, FAADMIN).
How to ensure that the roles and users in the Lightweight Directory Access Protocol (LDAP) store match what is available for selection when defining implementation users	Perform the Run User and Roles Synchronization Process task.
How to create a first implementation user	Prepare the IT Security Manager job role for user and role management so the super user and any other user provisioned with the IT Security Manager job role can manage users and roles.
How to establish security administration users	Define an IT security manager user provisioned with the IT Security Manager job role.
How to establish an implementation user with access to set up enterprise structures	Define an implementation user provisioned with the Application Implementation Consultant job role.

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

1. The Oracle Identity Management System Administrator user provisions the IT Security Manager job role with roles for user and role management.
2. The Oracle Fusion Applications super user synchronizes LDAP users with HCM user management so that users can be provisioned with roles through HCM.
3. The Oracle Fusion Applications super user performs the Create Implementation Users task to create one or more IT security manager and administrator users provisioned with security administrative entitlement.
4. The IT Security Manager user signs in to Oracle Fusion Applications and performs the Create Implementation Users task to create implementation managers and users.
5. The IT Security Manager user provisions implementation users for enterprise structure setup.

Note

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

Preparing the IT Security Manager Role

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

- Application Implementation Consultant
- IT Security Manager

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

- Identity User Administrators
- Role Administrators

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

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

This accesses the Welcome to Identity Manager Delegated Administration menu.

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

- In the Search Results, click the role's Display Name.
3. On the Hierarchy tab, select **Inherits From** and click **Add**.
 4. In the Add Parent Role to: IDENTITY USER ADMINISTRATORS window, select the role category: Common - Job Roles and add the IT Security Manager.

Click the arrow icon to show the list of available roles. Select IT Security Manager and move it to the **Roles to Add** list. Click **Save**.
 5. Search for the Role Administrators role, and repeat steps 1 to 4 to add that role to the IT Security Manager role's role inheritance.
 6. Assign the IT Security Manager role to the Xellerate Users organization.
 - a. In the Welcome to Identity Manager Delegated Administration menu (see step 1, above), in the Organizations list of tasks, click **Advanced Search - Organizations**.
 - b. Search for the Xellerate Users organization by entering Xellerate Users in **Display Name** and clicking **Search**.
 - c. In the Search Results, click the organization's Display Name. The Xellerate Users page appears.
 - d. Click the **Administrative Roles** link in the row of links above the Xellerate Users.
 - e. In **Filter By Role Name** of the Details window, enter the following string:


```
*IT_SECURITY_MANAGER*
```

Click **Find**.
 - f. Enable Read, Write, Delete, and Assign.
 - g. Click **Assign**.
 - h. Click **Confirm**.

Synchronizing Users and Roles from LDAP

Lightweight Directory Access Protocol (LDAP) must be synchronized with HCM user management so that users can be provisioned with roles through HCM.

1. Sign in to Oracle Fusion Applications using the super user's user name (for example FAADMIN) and password.

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

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

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

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

Defining an IT Security Manager User

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

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

The Oracle Identity Manager appears.

2. Click **Create User**.

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

3. Provide the following attributes:

Attribute	Value	Example
Last name	<any valid string>	Smith
Organization	Xellerate Users	N/A
User type	Non Worker	N/A
User login	<any valid string>	IT_SECURITY_MANAGER
Login password	<any valid string>	SeKur1TyPa\$\$w0Rd

Note

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

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

Defining an Implementation User for Enterprise Structures Setup

1. Sign in to Oracle Fusion Applications using the IT security manager user's name and password.
2. Create and provision an implementation user using the same task flow as for creating the IT security manager user in the previous section, except provision the following roles.
 - Application Implementation Manager

- Application Implementation Consultant

Note

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

Common Applications Configuration: Define Currencies and Currency Rates

Manage Currencies

Defining Currencies: Points to Consider

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

Currency Codes

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

Date Ranges

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

Symbols

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

Euro Currency Derivation: Explained

Use the Derivation Type, Derivation Factor, and Derivation Effective Date fields to define the relationship between the official currency (Euro) of the European Monetary Union (EMU) and the national currencies of EMU member states. For

each EMU currency, you define its Euro-to-EMU fixed conversion rate and the effective starting date.

Note

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

Derivation Type

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

Derivation Factor

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

Derivation Effective Date

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

FAQs for Manage Currencies

When do I create or enable currencies?

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

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

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

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

Note

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

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

What's a statistical unit currency type?

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

Manage Conversion Rate Types

Creating Conversion Rate Types: Critical Choices

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

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

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

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

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

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

- Maintain cross rate rules

Enforce Inverse Relationships

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

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

Select Pivot Currencies

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

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

Select Contra Currencies

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

Enable Cross Rates and Allow Cross Rate Overrides

Check the **Enable Cross Rates** check box to calculate conversion rates based on defined currency rate relationships. General Ledger calculates cross rates

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

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

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

Maintain Cross Rate Rules

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

Note

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

Using Rate Types: Examples

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

- Spot
- Corporate
- User
- Fixed

Scenario

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

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

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

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

Note

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

FAQs for Manage Conversion Rate Types

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

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

Rate Type	Usage
Spot	For currencies with fluctuating conversion rates or when exact currency conversion is needed.
Corporate	For establishment of a standard rate across your organization for a stable currency.
User	For infrequent entries where your daily rates for the entered foreign currency are not set up.

Fixed	For rates where the conversion is constant between two currencies.
-------	--

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

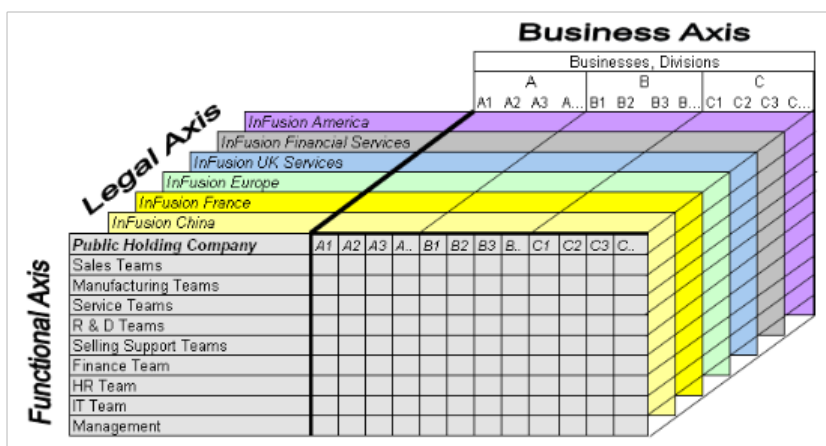
Common Applications Configuration: Define Enterprise Structures for Customer Data Management

Enterprise Structures: Overview

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

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

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



Legal Structure

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

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

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

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

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

Management Structure

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

Functional Structure

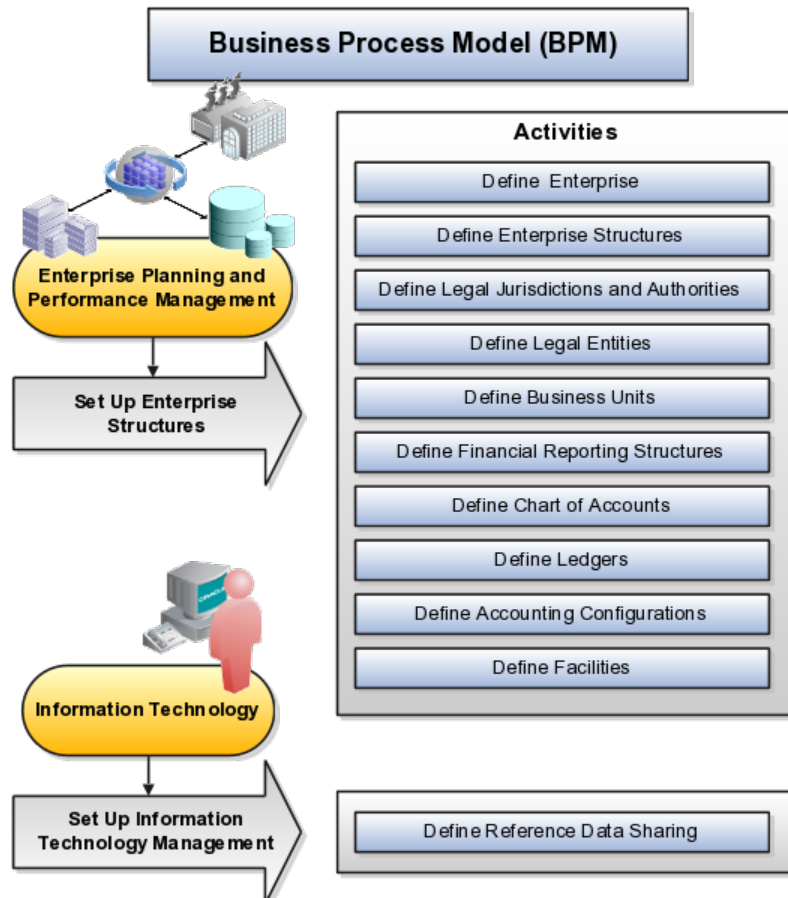
Straddling the legal and business organizations is a functional organization structured around people and their competencies. For example, sales,

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

Enterprise Structures Business Process Model: Explained

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

The following figure and chart describes the Business Process Model structures and activities.



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

Note

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

Global Enterprise Configuration: Points to Consider

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

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

Enterprise Configuration

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

Business Unit Management

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

Security Structure

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

Compliance Requirements

How do you comply with your corporate external reporting requirements and local statutory reporting requirements? Do you tend to prefer a corporate first or

an autonomous local approach? Where are you on a spectrum of centralization, very centralized or decentralized?

Modeling Your Enterprise Management Structure in Oracle Fusion: Example

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

Scenario

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

InFusion Corporation

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

Analysis

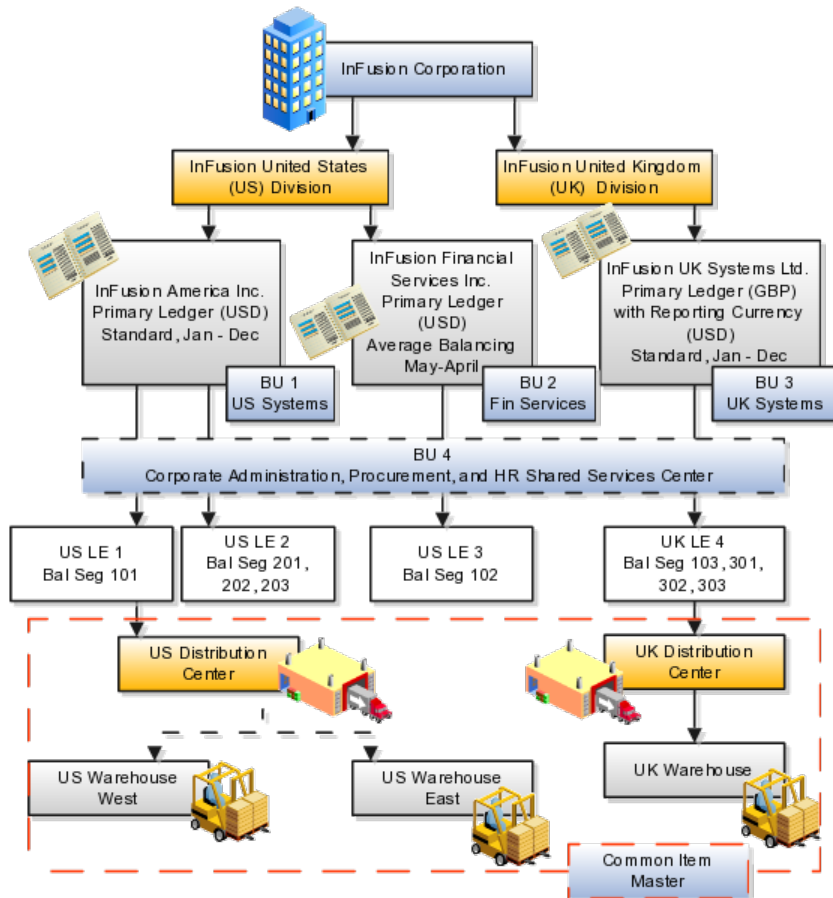
The following are elements you need to consider in creating your model for your global enterprise structure.

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

Global Enterprise Structure Model

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

- Creation of three separate ledgers representing your separate legal entities:
 - InFusion America Inc.
 - InFusion Financial Services Inc.
 - InFusion UK Services Ltd.
- Consolidation of results for system components, installations, and maintenance product lines across the enterprise
- All UK general and administrative costs processed at the UK headquarters
- US Systems' general and administrative costs processed at US Corporate headquarters
- US Financial Services maintains its own payables and receivables departments



Real World Entity	Entity Name	Enterprise	Legal Entity	BSV	Ledger	BU	Cost Center	Dept	Item Master Org	Inventory Org
Enterprise	InFusion Group	M								
Company	US LE 1		M	O	O					
Company	US LE 2		M	O	O					
Company	US LE 3		M	O	O					
Company	UK LE 4		M	O	O					
Business Unit	US Systems BU 1					M				
Business Unit	FIN Services BU 2					M				
Business Unit	UK Systems BU 3					M				
Division	InFusion UK			O			O			
Division	InFusion US			O			O			
Headquarters	BU 4					M				
Shared Service Center	BU 4					M				
Department	AP Department						O	M		
List of Items	Common Item Master								O	
Distribution Center	US Distribution Center									O
Distribution Center	UK Distribution Center									O
Warehouse	US Warehouse West									O
Warehouse	US Warehouse East									O
Warehouse	UK Warehouse									O
BSV = Balancing Segment Value										
BU = Business Unit										
Dept = Department										
Org = Organization										
M = Mandatory Setup										
O = Optional Setup										

In this chart, the green globe stands for mandatory and gold globe stands for optional setup. The following statements expand on the data in the chart.

- The enterprise is mandatory because it serves as an umbrella for the entire implementation. All organizations are created within an enterprise.
- Legal entities are also mandatory. They can be optionally mapped to balancing segment values or represented by ledgers. Mapping balancing segment values to legal entities is mandatory if you plan to use the intercompany functionality.
- At least one ledger is mandatory in an implementation in which you record your accounting transactions.
- Business units are also mandatory because financial transactions are processed in business units.
- A shared service center is optional, but if used, must be a business unit.
- Divisions are optional and can be represented with a hierarchy of cost centers or by a second balancing segment value.
- Departments are mandatory because they track your employees.
- Optionally, add an item master organization and inventory organizations if you are tracking your inventory transactions in Oracle Fusion Applications.

Note

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

Note

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

Essbase Character and Word Limitations

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

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

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

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

Restricted Characters

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

Character	Meaning
@	at sign
\	backslash
,	comma
-	dash, hyphen, or minus sign
=	equal sign
<	less than sign
()	parentheses

.	period
+	plus sign
'	single quotation mark
_	underscore
	vertical bar

Other Restrictions

- Do not place spaces at the beginning or end of names. Essbase ignores such spaces.
- Do not use these types of words as dimension or member names:
 - Calculation script commands, operators, and keywords.
 - Report writer commands.
 - Function names and function arguments.
 - Names of other dimensions and members (unless the member is shared).
 - Generation names, level names, and aliases in the database.
 - Any of these words in the table below:

List 1	List 2	List 3
ALL	AND	ASSIGN
AVERAGE	CALC	CALCMBR
COPYFORWARD	CROSSDIM	CURMBRNAME
DIM	DIMNAME	DIV
DYNAMIC	EMPTYPARM	EQ
EQOP	EXCEPT	EXP
EXPERROR	FLOAT	FUNCTION
GE	GEN	GENRANGE
GROUP	GT	ID
IDERROR	INTEGER	LE
LEVELRANGE	LOOPBLOCK	LOOPPARMS
LT	MBR	MBRNAME
MBRONLY	MINUS	MISSING, #MISSING
MUL	MULOP	NE
NON	NONINPUT	NOT
OR	PAREN	PARENPARM
PERCENT	PLUS	RELOP
SET	SKIPBOTH	SKIPMISSING
SKIPNONE	SKIPZERO	TO

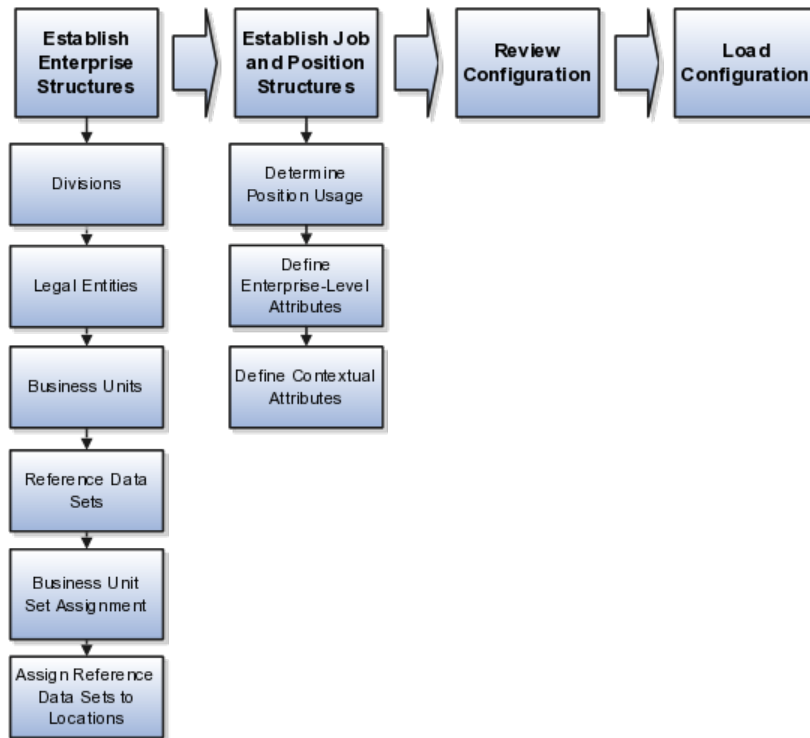
TOLOCALRATE	TRAILMISSING	TRAILSUM
UMINUS	UPPER	VARORXMBR
XMRONLY	\$\$\$UNIVERSE\$\$\$	#MI

Define Initial Configuration with the Enterprise Structures Configurator

Establishing Enterprise Structures Using the Enterprise Structures Configurator: Explained

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

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



To be able to use the Enterprise Structures Configurator, you must select the Enterprise Structures Guided Flow feature for your offerings on the Configure Offerings page in the Setup and Maintenance work area. If you do not select

this feature, then you must set up your enterprise structure using individual tasks provided elsewhere in the offerings, and you cannot create multiple configurations to compare different scenarios.

Establish Enterprise Structures

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

Establish Job and Position Structures

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

Review Configuration

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

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

- Legislative data groups (the application creates one legislative data group for each country that is identified in the configuration.)
- Name of the legislative data group that will be assigned to the payroll statutory unit that is generated for each legal entity.
- Organization hierarchy.

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

Note

If your PDF viewer preferences are set to open PDFs in a browser window, the Technical Summary report replaces the Oracle Fusion application. Use your browser's Back button to return to the application.

Load Configuration

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

Using Rollback for an Enterprise Structures Configuration: Explained

The Enterprise Structures Configurator provides the ability to roll back, or undo, an enterprise configuration. Two methods for rolling back a configuration are available: manual rollback, and automatic rollback.

Manual Rollback

Use the manual method for rolling back an enterprise configuration when you have loaded a configuration, but then decide you do not want to use it.

Automatic Rollback

The automatic rollback is used when you run the Load Configuration process, but the process encounters an error. In this case, the application rolls back any enterprise structures that were created before the error was encountered.

Designing an Enterprise Configuration: Example

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

Scenario

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

Enterprise Details

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

Analysis

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

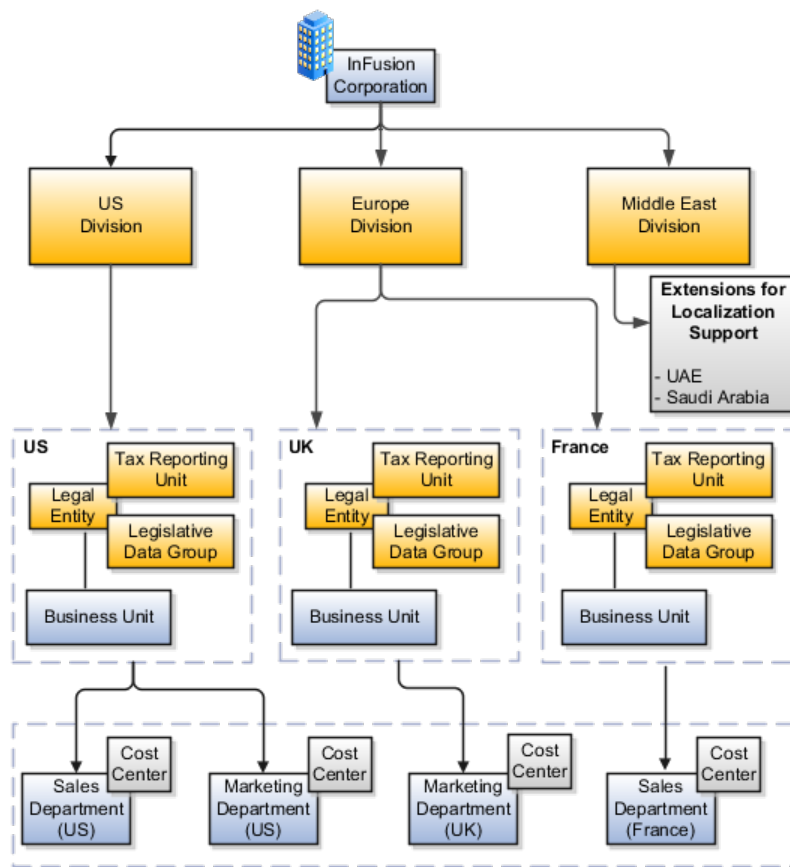
InFusion requires legal entities with legal employers, payroll statutory units, tax reporting units, and legislative data groups for the US, UK, France, Saudi Arabia, and UAE, in order to employ and pay its workers in those countries.

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

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

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

This figure illustrates the enterprise configuration that results from the analysis of InFusion Corporation.



Division: Explained

Managing multiple businesses requires that you segregate them by their strategic objectives and measure their results. Responsibility to reach objectives can be delegated along the management structure. Although related to your legal structure, the business organizational hierarchies do not need to reflect directly

the legal structure of the enterprise. The management entities and structure can include divisions and subdivisions, lines of business, and other strategic business units, and include their own revenue and cost centers. These organizations can be included in many alternative hierarchies and used for reporting, as long as they have representation in the chart of accounts.

Divisions

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

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

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

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

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

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

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

Legal Entities: Explained

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

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

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

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

The Role of Your Legal Entities

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

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

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

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

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

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

Using the Enterprise Structures Configurator (ESC), you can create legal entities for your enterprise automatically, based on the countries in which divisions

of your business operate, or you can upload a list of legal entities from a spreadsheet.

Automatically Creating Legal Entities

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

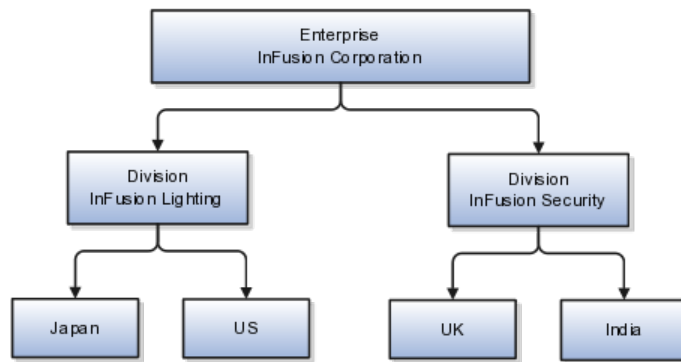
The application automatically creates the legal entities that you select, and identifies them as payroll statutory units and legal employers. For each country that you indicated that your enterprise operates in, and for each country that you created a location for, the application also automatically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

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

Example: Creating Legal Entities Automatically

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

This figure illustrates InFusion Corporation's enterprise structure.



This table represents the selections that InFusion Corporation makes when specifying which legal entities to create on the Map Divisions by Country page.

Country	Enterprise	InFusion Lighting	InFusion Security
Japan	No	Yes	No
US	No	Yes	No

UK	No	No	Yes
India	No	No	Yes

Based on the selections made in the preceding table, the ESC creates the following four legal entities:

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

Creating Legal Entities Using a Spreadsheet

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

Legal Entity in Oracle Fusion: Points to Consider

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

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

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

The Importance of Legal Entity in Transactions

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

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

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

Legal Entity and Its Relationship to Business Units

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

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

Legal Entity and Its Relationship to Divisions

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

Legal Entity and Its Relationship to Ledgers

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

Legal Entity and Its Relationship to Balancing Segments

Oracle Fusion General Ledger supports up to three balancing segments. Best practices recommend that one of these segments represents your legal entity to

ease your requirement to account for your operations to regulatory agencies, tax authorities, and investors. Accounting for your operations means you must produce a balanced trial balance sheet by legal entity. If you account for many legal entities in a single ledger, you must:

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

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

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

Note

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

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

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

Note

While transactions that cross balancing segments don't necessarily cross legal entity boundaries, all transactions that cross legal entity boundaries must cross balancing segments. If you make an acquisition or are preparing to dispose of a portion of your enterprise, you may want to account for that part of the enterprise in its own balancing segment even if it is not a separate legal entity.

If you do not map legal entities sharing the same ledger to balancing segments, you will not be able to distinguish them using the intercompany functionality or track their individual equity.

Legal Entity and Its Relationship to Consolidation Rules

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

Legal Entity and its Relationship to Intercompany Transactions

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

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

Note

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

Legal Entity and Its Relationship to Worker Assignments and Legal Employer

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

Legal Entity and Payroll Reporting

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

Business Units: Explained

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

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

The Oracle Fusion Applications business unit model:

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

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

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

In summary, use business units in the following ways:

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

Brief Overview of Business Unit Security

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

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

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

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

Automatically Creating Business Units

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

You can automatically create business units at the following levels:

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

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

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

After you select a business unit level, the application generates a list of business units, and you select the ones you want the application to create. If you select a

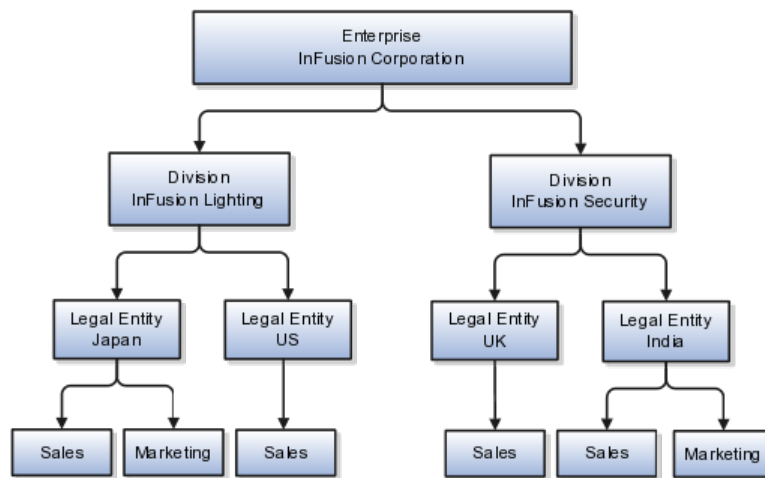
level that has two components, such as country and division, then the system displays a table listing both components, and you select the check boxes at the intersections of the components.

The business units listed by the application are suggestions only, and are meant to simplify the process to create business units. You are not required to select all of the business units suggested. When you navigate to the next page in the ESC guided flow, which is the Manage Business Units page, you cannot delete any of the business units that were created automatically. You must return to the Create Business Units page and deselect any business units that you no longer want.

Example: Selecting Business Unit Levels

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

This figure illustrates InFusion Corporation's enterprise structure.



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

Business Unit Level	Suggested Business Units
Country	<ul style="list-style-type: none"> • US • UK • Japan • India
Country and Division	<ul style="list-style-type: none"> • InFusion Lighting: Japan • InFusion Lighting: US • InFusion Security: UK • InFusion Security: India

Country and business function	<ul style="list-style-type: none"> • Sales: Japan • Marketing: Japan • Sales: US • Sales: UK • Marketing: India • Sales: India
Division	<ul style="list-style-type: none"> • InFusion Lighting • InFusion Security
Division and Legal Entity	<ul style="list-style-type: none"> • InFusion Lighting: Japan • InFusion Lighting: US • Infusion Security: UK • Infusion Security: India
Division and Business Function	<ul style="list-style-type: none"> • InFusion Lighting, Sales • InFusion Lighting, Marketing • InFusion Security, Sales • InFusion Security, Marketing
Business Function	<ul style="list-style-type: none"> • Sales • Marketing
Legal Entity	<ul style="list-style-type: none"> • Legal Entity: Japan • Legal Entity: US • Legal Entity: UK • Legal Entity India
Legal Entity and Business Function	<ul style="list-style-type: none"> • Legal Entity: Japan, Sales • Legal Entity: Japan, Marketing • Legal Entity: US, Sales • Legal Entity: UK, Sales • Legal Entity India, Marketing • Legal Entity India, Sales

Manually Creating Business Units

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

Reference Data Sets and Sharing Methods: Explained

Oracle Fusion Applications reference data sharing feature is also known as SetID. The reference data sharing functionality supports operations in multiple

ledgers, business units, and warehouses, thereby reducing the administrative burden and decreasing the time needed to implement new business units. For example, you can share sales methods, transaction types, or payment terms across business units or selected other data across asset books, cost organizations, or project units.

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

Reference Data Sets

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

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

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

Reference Data Sharing Methods

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

- Assignment to one set only, no common values allowed. The simplest form of sharing reference data that allows assigning a reference data object instance to one and only one set. For example, Asset Prorate Conventions are defined and assigned to only one reference data set. This set can be shared across multiple asset books, but all the values are contained only in this one set.
- Assignment to one set only, with common values. The most commonly used method of sharing reference data that allows defining reference data object instance across all sets. For example, Receivables Transaction Types are assigned to a common set that is available to all the business units without the need to be explicitly assigned the transaction types to each business unit. In addition, you can assign a business unit specific set of transaction types. At transaction entry, the list of values for transaction

types includes transaction types from the set assigned to the business unit, as well as transaction types assigned to the common set that is shared across all business units.

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

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

Business Units and Reference Data Sets: How They Work Together

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

Common Set Versus Specific Sets

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

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

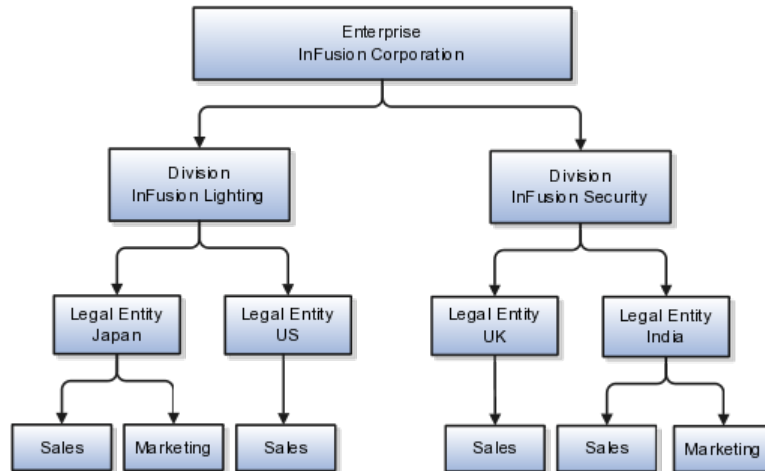
Business Unit Set Assignment

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

Example: Assigning Sets to Business Units

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

The following figure illustrates the structure of InFusion Corporation.



When deciding how to create business units, InFusion decides to create them using the country and business function level. Therefore, they created the following business units:

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

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

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

InFusion assigns business units to sets as follows:

Business Unit	Default Set Assignment	Set Assignment Overrides
Sales_Japan	Sales_Japan_Set for grades, departments, and locations	Jobs set for jobs

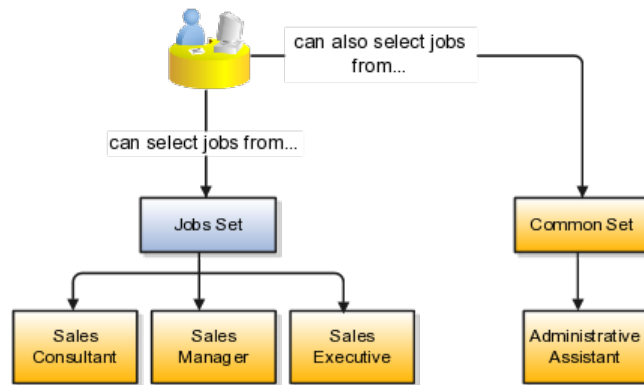
Marketing_Japan	Mktg_Japan_Set for grades, departments, and locations	None
Sales_US	Sales_US_Set for grades, departments, and locations	Jobs set for jobs
Sales_UK	Sales_UK_Set for grades, departments, and locations	Jobs set for jobs
Marketing_India	Mktg_India_Set for grades, departments, and locations	None
Sales_India	Sales_India_Set for grades, departments, and locations	Jobs set for jobs

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

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

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

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



Creating Reference Data Sets in the Enterprise Structures Configurator: Explained

If you created business units automatically, then the Enterprise Structures Configurator automatically creates reference data sets for you. The Enterprise Structures Configurator creates one reference data set for each business unit. You

can add additional sets, but you cannot delete any of the sets that were created automatically.

A standard set called the Enterprise set is predefined.

Common Set

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

Jobs and Positions: Critical Choices

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

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

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

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

Primary Industry of Your Enterprise

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

Primary Industry	Workforce Setup
Mining	Positions
Utilities	Positions
Manufacturing	Positions
Retail Trade	Positions
Transportation and Warehousing	Positions
Educational Services	Positions
Public Transportation	Positions
Agriculture, Forestry, Fishing, and Hunting	Jobs

Construction	Jobs
Wholesale Trade	Jobs
Information	Jobs
Finance and Insurance	Jobs
Professional, Scientific, and Technical Services	Jobs
Management of Companies and Enterprises	Jobs
Administrative and Support and Waste Management and Remediation Services	Jobs
Arts, Entertainment, and Recreation	Jobs
Accommodation and Food Services	Jobs
Other Services (Except Public Administration)	Jobs

Management of People

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

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

Positions: Examples

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

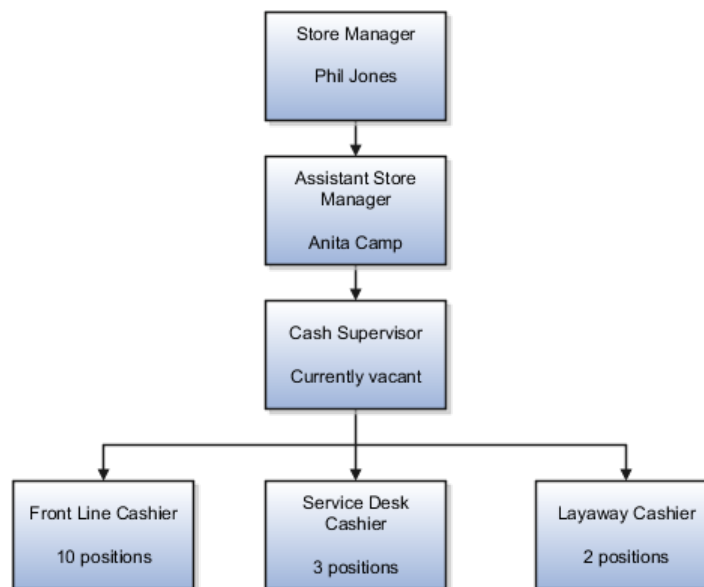
Retail Industry

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

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

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

This figure illustrates the retail position setup.

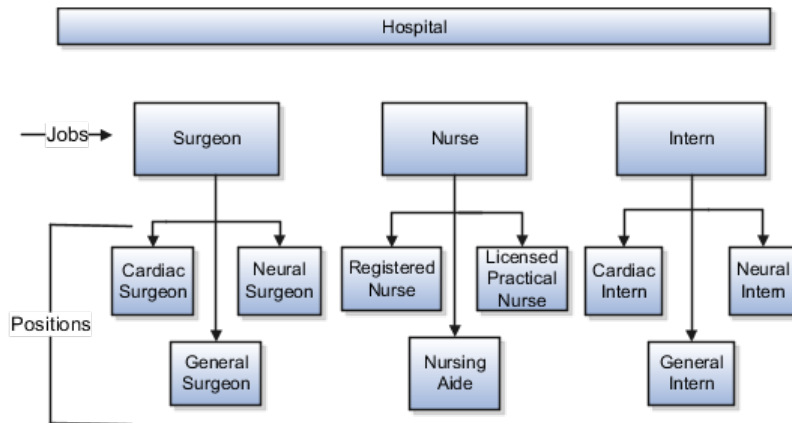


Health Care Industry

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

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

This figure illustrates the hospital position setup.



Jobs: Example

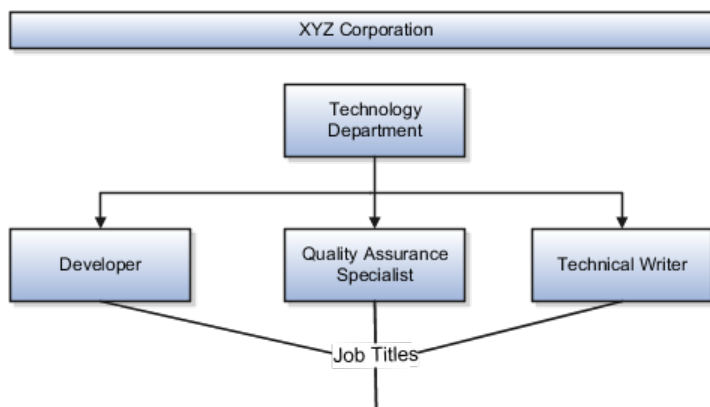
Jobs are typically used without positions by service industries where flexibility and organizational change are key features.

Software Industry

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

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

This figure illustrates the software industry job setup.



Job and Position Structures: Explained

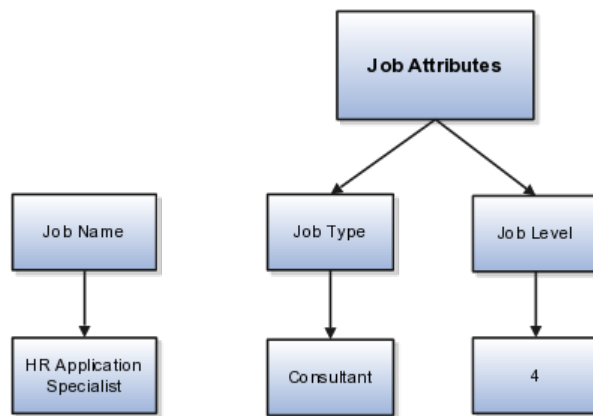
Job and position structures identify the descriptive flexfield structure that enables you to specify additional attributes that you want to capture when you

define jobs and positions. Job and position attributes provide further detail to make jobs and positions more specific. You also use attributes to define the structure of your jobs and positions. You can specify attributes at the enterprise level for jobs and positions, at the business unit level for positions, and at the reference data set level for jobs. Job and position structures are optional.

Enterprise-Level Job Attributes

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

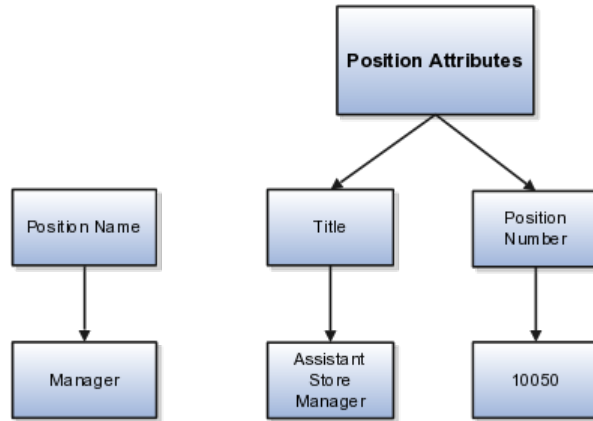
This figure illustrates how job type and job level provide further details for the HR Application Specialist job.



Enterprise-Level Position Attributes

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

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



Business Unit-Level Attributes for Positions

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

Reference Data Set-Level Attributes for Jobs

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

FAQs for Define Initial Configuration

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

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

What's an ultimate holding company?

The legal entity that represents the top level in your organization hierarchy, as defined by the legal name entered for the enterprise. This designation is used

only to create an organization tree, with the ultimate holding company as the top level, divisions and country holding companies as the second level, and legal employers as the third level.

What's the default reference data set?

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

What happens if I override the set assignment?

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

Define Reference Data Sharing

Reference Data Sharing: Explained

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

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

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

Note

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

Reference Data Sets: Explained

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

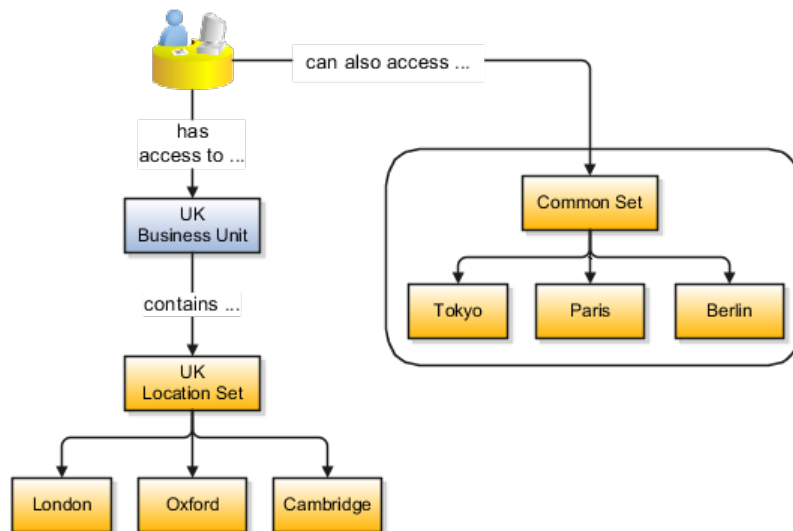
Consider the following scenario.

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

Partitioning

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

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



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.

Type	Description
Asset Book	Information about the acquisition, depreciation, and retirement of an asset that belongs to a ledger or a business unit.
Business Unit	The departments or organizations within an enterprise.
Cost Organization	The organization used for cost accounting and reporting on various inventory and cost centers within an enterprise.
Project Unit	A logical organization within an enterprise that is responsible for enforcing consistent project management practices.

Reference Data Set	References to other shared reference data sets.
--------------------	---

Determinant

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

Reference Groups

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

Note

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

Define Enterprise: Manage Enterprise HCM Information

Enterprise: Explained

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

Enterprise Defined

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

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

Oracle Fusion Applications offers capabilities for multiple tenants to share the same applications instance for some human resources processes. If you offer

business process outsourcing services to a set of clients, each of those clients may be represented as an enterprise within an Oracle Fusion Application instance. To support this functionality, system owned reference data such as sequences, sets, and flexfields are also defined within an enterprise.

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

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

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

Define Enterprise: Manage Locations

Locations: Explained

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

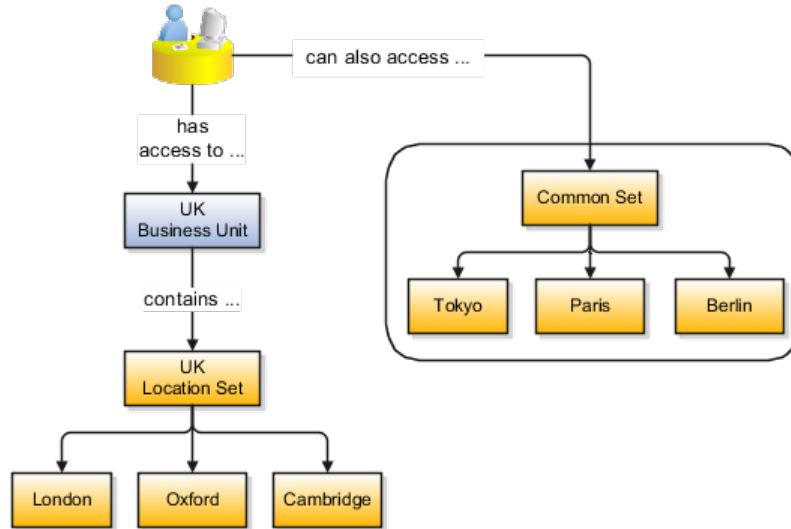
The locations that you create exist as separate structures that you can use for reporting purposes, and also in rules that determine employee eligibility for various types of compensation and benefits. You enter information about a location only once. Subsequently, when you set up other workforce structures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

Location Sets

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

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

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



Creating Multiple Locations Simultaneously

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

FAQs for Manage Locations

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

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

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

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

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

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

What happens if I inactivate a location?

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

How can I associate a location with an inventory organization?

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

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

Define Geographies

Defining Address Cleansing: Explained

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

Address cleansing can only be used through the Oracle Fusion Trading Community Data Quality product, because the feature is delivered using Data Quality integration. You need to ensure that you have a license for the countries that will use Trading Community Data Quality data cleansing.

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

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

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

Geography Structure

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

Geography Hierarchy

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

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

Geography Validation

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

Geography Structures: Explained

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

You can use the geography structure to establish:

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

How Geographies Can Be Related

You can determine how a country's geographies are hierarchically related by creating the hierarchy of the geography types in the geography structure. When

you define a country's structure the country geography type is implicitly at the top of the geography structure, and the numbering of the subsequent levels start with 1 as the next geography level after country.

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

Note

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

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

The Types of Geographies You Can Define for the Country

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

Note

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

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

Geography Hierarchy: Explained

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

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

The application leverages geography hierarchy information to facilitate business processes that rely on geography information, for example, tax calculation, order sourcing rules, sales territory definition. The geography hierarchy information

is centrally located in the Trading Community Model and shared among other application offerings.

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

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

Geography

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

Geography Type

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

Geography Usage

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

Master Reference Geography Hierarchy

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

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

User Defined Zones

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

Geography Validation: Explained

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

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

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

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

Map to Attribute

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

Note

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

Enable List of Values

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

Tax Validation

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

Geography Validation

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

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

Note

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

Geography Validation Control

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

These are the geography validation levels you can choose:

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

Regardless of the result of validation, the validation process will try to map any address attribute to a geography of the country, and store any mapping

it could establish based on the available data. This is called **Geography Name Referencing** and it is executed as part of validation. The result of this referencing is used in several business processes in the application to map an address to a specific geography or zone.

Note

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

Importing Geographies: Explained

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

Consider the following when importing geographies:

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

File-Based Import Option

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

Geography Loader Process Option

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

Import Object Entity, Interface Table, and Destination Tables

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

Note

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

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

File-Based Import Entities	Interface Tables	Destination Tables	Application Object
ImpGeography	HZ_IMP_GEOGRAPHIES	HZ_GEOGRAPHIES HZ_GEOGRAPHY_IDEN HZ_GEOGRAPHY_TYPE HZ_HIERARCHY_NODE	Geography

Importing Country Structures Using File-Based Import: Explained

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

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

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

Consider the following questions when importing your data:

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

Comparing Business Object Structures

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

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

Import Objects for the Country Structure

To facilitate the import of country structures, Oracle Fusion Applications incorporate the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

Comparing Business Object Data

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

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

Import Object	Related Import Object Topic
Country Structure	Country Structure Import Objects: How They Work Together

Extensible Attributes

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

Importing Country Structures Using File-Based Data Import

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

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

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

Verifying Your Imported Data

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

Country Structure Import Objects: How They Work Together

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

- Target import object concepts
- Target objects for the Country Structure import object
- Target import object attributes

- Target object attribute reference guide files

Target Import Object Concepts

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

Country Structure Target Import Objects

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

Target Import Objects Attributes

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

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

Note

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

Target Import Objects Attributes Resources

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

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

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

Target Import Object	Description	Reference Guide File Names
ImpGeoStructureLevel	<p>Contains information that specifies a country's geography structure.</p> <p>Sample attributes: GeographyType, GeographyTypeName, LevelNumber, and ParentGeographyType.</p> <p>Reference attribute: CountryCode</p>	HZ_IMP_GEO_STRUCTURE_LEVELS_Reference

Importing Geographies Using File-Based Import: Explained

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

Consider the following questions when importing your data:

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

Comparing Business Object Structures

You must understand how your geography data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a geography.

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

Import Objects for the Geography

To facilitate the import of geographies, Oracle Fusion Applications incorporate the structure of the geography into import objects. The import object for the geography is ImpGeography.

Comparing Business Object Data

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

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

Import Object	Related Import Object Topic
ImpGeography	Geography Import Objects: How They Work Together

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

Extensible Attributes

Oracle Fusion Applications do not support extensible attributes for geographies. You can only import data for attributes provided by Oracle Fusion Applications.

Importing Geographies Using File-Based Data Import

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

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

When importing geography information, you must provide the parent reference information for all parent levels for the entity.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. Users with the Master Data Management

Administrator job role can also navigate to the Manage Geographies work area to view the imported geographies.

Geography Import Objects: How They Work Together

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

This topic introduces the following:

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

Target Import Object Concepts

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

Geography Target Import Objects

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

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

Note

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

Target Import Objects Attributes

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

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can

predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Target Import Objects Attributes Resources

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

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

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

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

Importing Geographies Using File-based Data Import: Worked Example

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

The following table summarizes the key decisions for this scenario:

Decisions to Consider	In This Example
What type of object are you importing?	Geography
What file type are you using for your source data?	Text file
Where are you uploading your source data file from?	Your desktop
What data type is your source data file?	Comma separated
Which fields are you importing into Oracle Fusion applications?	All, except for the RecordTypeCode field

When do you want to process the import?	Immediately
---	-------------

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

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

Prerequisites when importing additional geography data after your initial import

1. You need to ensure that the combination of Source ID and Parent Source ID values are unique for each row of data within a single import. However, your source data files do not need to have the same Source ID and Parent Source ID values as your previously imported geography data. If the geography structure levels and the parents for each geography value are the same, the changed IDs will not affect the import.
2. Ensure that all of the parents of a child geography are included in your data file so that the child geography can be added. For example, if you originally imported US, CA, and San Francisco, and now you want to import the city of San Jose in CA, then your data file needs to include US, CA, and San Jose.
3. Check that your source data file has the correct values for the geography data that you have already loaded. For example, if your initial import included the value US for country and CA as state, and in a subsequent import you have California as a state, your geography import will result in two state records (CA and California) in the application data, with the US as the country parent.

Determine what information is in the source file

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

Geography Level	Name	Source ID	Parent Source ID
1 (Country)	US	1	
2 (State)	CA	11	1
3 (County)	Alameda	111	11
4 (City)	Pleasanton	1111	111
4 (City)	Dublin	1112	111

Create and schedule the import activity

You create an import activity, enter the import details, and schedule the import. An import activity definition provides the instructions for the import processing

- this includes selecting the source file, or file location; mapping fields from the source file to the Oracle Fusion object and attribute; and scheduling the import.

1. Navigate to Setup and Maintenance and search for the Manage File Import Activities task. Click **Go to Task**.
2. In the Manage Import Activities page, click the **Create** icon.
3. In the Create Import Activity: Set Up page, create an import activity for the Geography object type by completing the fields, as shown in this table:

Field	Value
Name	Master Reference Geographies
Object	Geography
File Type	Text File
File Selection	Specific file
Upload From	Desktop
File Name	Choose relevant file from desktop
Data Type	Comma separated

Note

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

4. Click **Next**.
5. On the Create Import Activity: Map Fields page, map each field from your source file to the Oracle Fusion object and attribute, as shown in this example:

Column Header	Example Value	Ignore	Object	Attribute
Primary Geography Name	Primary Geography Name	United States	Imp Geography	Primary Geography Name
Country Code	US	No	Imp Geography	Country Code
Record Type Code	0	Yes	Imp Geography	Record Type Code
Source ID	10265	No	Imp Geography	Source ID
Parent Source ID	1053	No	Imp Geography	Parent Source ID

If you do not want to import a column in the text file you can select **Ignore**.

Note

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

6. Click **Next**.
7. On the Create Import Activity: Create Schedule page, select **Immediate** in the Schedule field so that the import will start immediately.

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

8. Click **Next**.

Monitor the import results

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

1. On the Create Import Activity: Review and Activate page, you verify your import details in the Import Details, File Details, Import Options, and Schedule sections.
2. Your import details are correct so you click **Activate** to submit the import.

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

Importing and Exporting Territory Geography Zones: Explained

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

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

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

Import the master reference geography data

Firstly, you need to import the master reference geography data. Master reference geography data consists of geography elements such as country, state,

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

Define your territory geography zones

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

Export the territory geography zones

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

Import the territory geography zones

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

Note

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

Managing Geography Structures, Hierarchies, and Validation: Worked Example

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

The following table summarizes the key decisions for this scenario.

Decisions to Consider	In This Example
Copy an existing country structure?	No, create a new country structure.
What is the structure of the geography types?	Create geography types with the following ranking structure: <ol style="list-style-type: none"> 1. County 2. Post Town

What is the geography hierarchy?	Create the following hierarchy: <ol style="list-style-type: none"> 1. Country of United Kingdom 2. County of Berkshire 3. Post Town of Reading
Which address style format will you use when mapping geography validations?	The default address style format, called the No Styles Format.
Are you using Oracle Fusion Tax for tax purposes?	No, do not select Tax Validation for the geography types.

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

Defining the geography structure

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

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

Defining the geography hierarchy

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

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

8. Click the **Create** button.
9. In the **Create Post Town** page, Primary and Alternate Names section, enter Reading in the **Name** field.
10. Click Save and Close.

Defining the geography validations

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

1. On the Manage Geographies page, click **Validation Defined**.
2. On the Manage Geography Validation page, Address Style section, click **No Styles Format** to highlight the table row.
3. For the County geography type, click the **County** list item in the **Map to Attribute** field.
4. Click the **Enable List of Values** option for the County geography type.
5. Click the **Geography Validation** option for the County geography type.
6. For the Post Town geography type, click the **City** list item in the **Map to Attribute** field.
7. Click the **Geography Validation** option for the Post Town geography type.
8. In the Geography Validation Control section, click the **Error** list item in the **Geography Validation Level for Country** field.
9. Click **Save and Close**.

FAQs for Define Geographies

When do I define address cleansing?

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

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

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

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

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

States country geography structure, then the State level cannot be deleted in the country geography structure.

Can I add any geography to the geography hierarchy?

Yes. However, the geography type for the geography that you want to add must be already added to the country geography structure.

Can I edit a specific geography in the geography hierarchy?

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

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

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

Define Legal Entities for Customer Data Management: Manage Legal Entity

Legal Entities: Explained

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

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

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

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

The Role of Your Legal Entities

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

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

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

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

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

Legal Entity in Oracle Fusion: Points to Consider

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

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

- The importance of legal entity in transactions
- Legal entity and its relationship to business units

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

The Importance of Legal Entity in Transactions

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

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

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

Legal Entity and Its Relationship to Business Units

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

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

Legal Entity and Its Relationship to Divisions

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

Legal Entity and Its Relationship to Ledgers

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

Legal Entity and Its Relationship to Balancing Segments

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

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

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

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

Note

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

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

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

Note

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

Legal Entity and Its Relationship to Consolidation Rules

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

Legal Entity and its Relationship to Intercompany Transactions

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

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

Note

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

Legal Entity and Its Relationship to Worker Assignments and Legal Employer

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

Legal Entity and Payroll Reporting

Your legal entities are required to pay payroll tax and social insurance such as social security on your payroll. In Oracle Fusion Applications, you can register

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

Define Legal Entities for Customer Data Management: Manage Legal Entity HCM Information

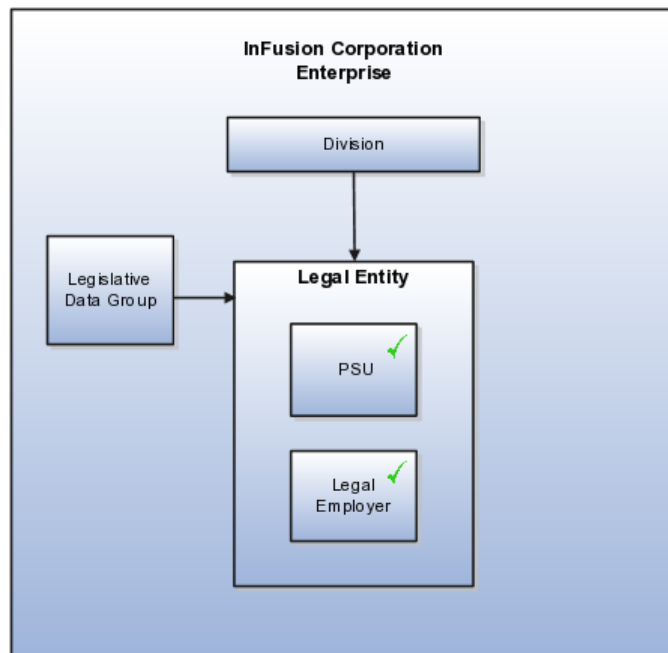
HCM Organization Models: Examples

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

Simple Configuration

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

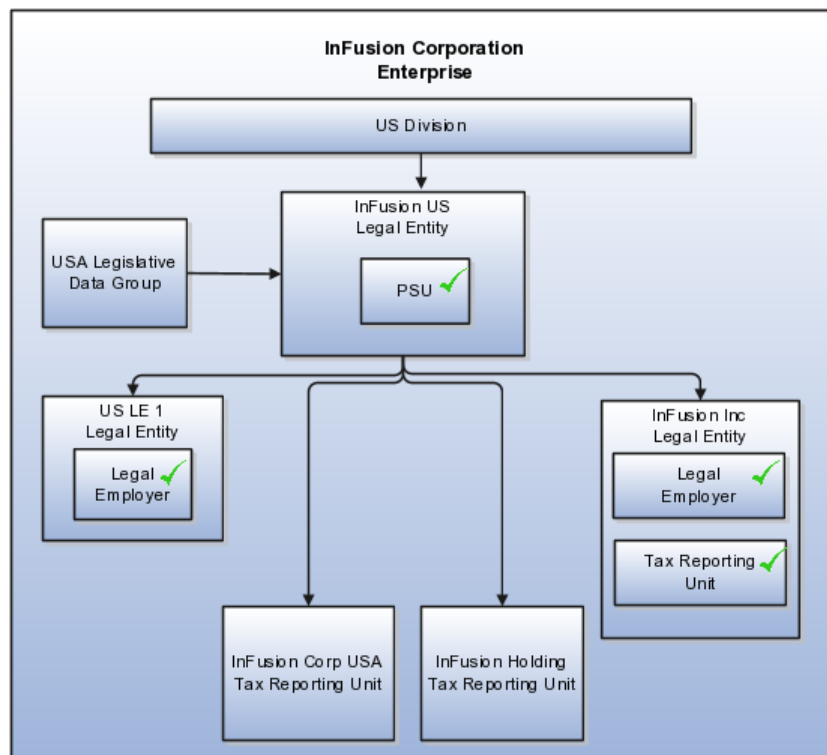
This figure illustrates a simple configuration where the enterprise has only one legal entity that is both a payroll statutory unit and a legal employer.



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

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

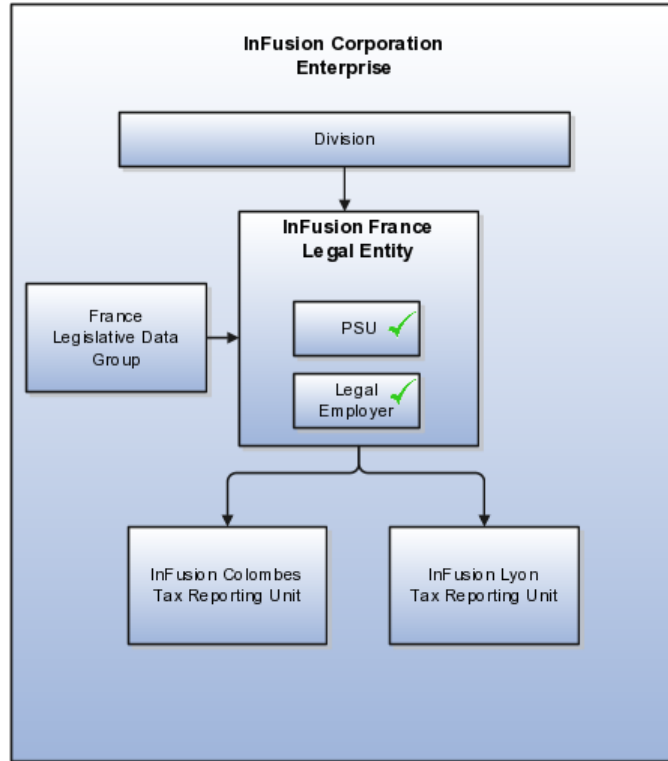
This figure illustrates an enterprise that has one payroll statutory unit and multiple legal employers and tax reporting units.



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

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

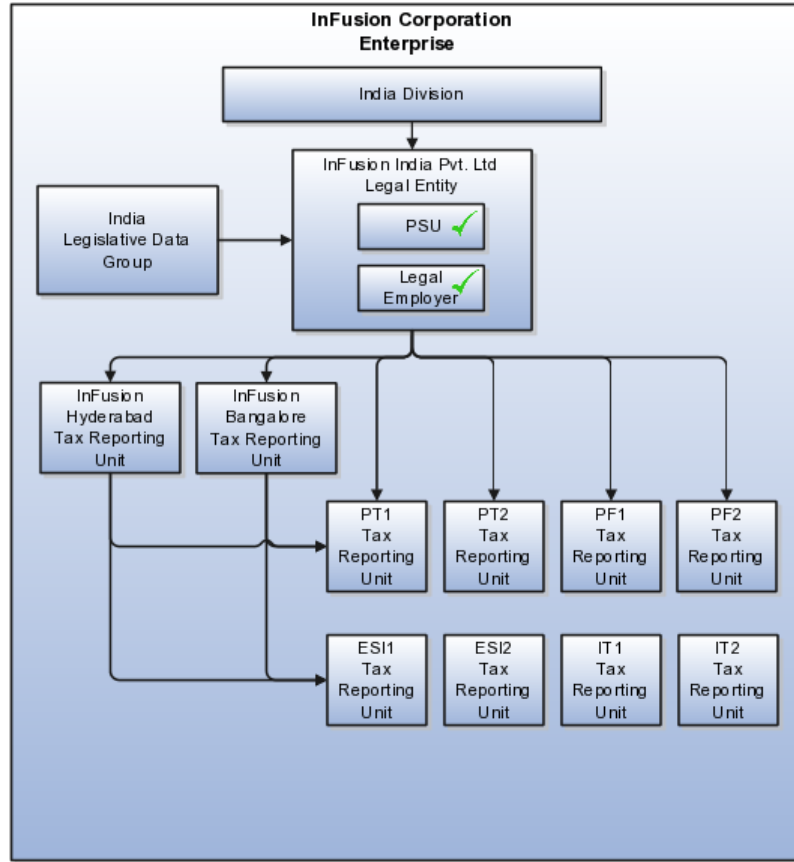
This figure illustrates an example of an organization with one legal entity that is both a legal employer and a payroll statutory unit and that has two tax reporting units.



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

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

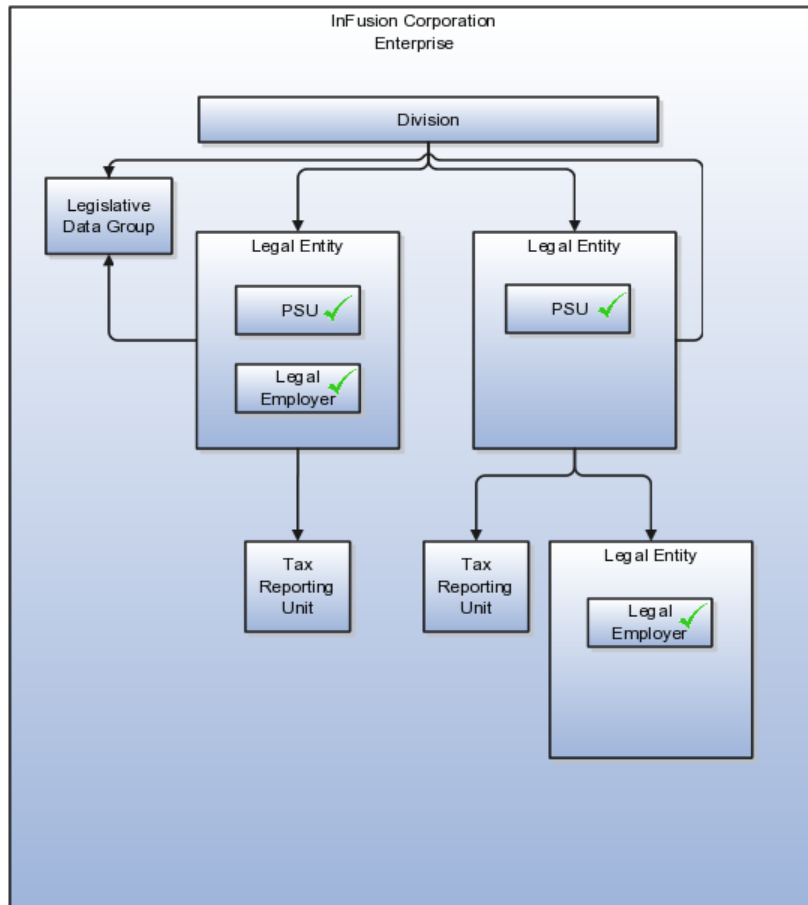
This figure illustrates an enterprise with one legal entity that is a payroll statutory unit and a legal employer, and the tax reporting units are independent from the legal employer.



Multiple Payroll Statutory Units with Several Tax Reporting Units that are Independent from the Legal Employer

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

This figure illustrates an enterprise with two legal entities, and legal employers and tax reporting units are independent from each other.



Payroll Statutory Units, Legal Employers, and Tax Reporting Units: How They Work Together

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

Legal Employers and Payroll Statutory Unit

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

Payroll Statutory Units and Tax Reporting Units

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

Tax Reporting Units and Legal Employers

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

FAQs for Manage Legal Entity HCM Information

What's a legal employer?

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

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

What's a payroll statutory unit?

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

Define Business Units for Customer Data Management: Manage Service Provider Relationships

Shared Service Centers: Explained

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

Service Provider Model

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

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

Shared Service Center: Points to Consider

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

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

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

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

Security

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

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

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

Benefits

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

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

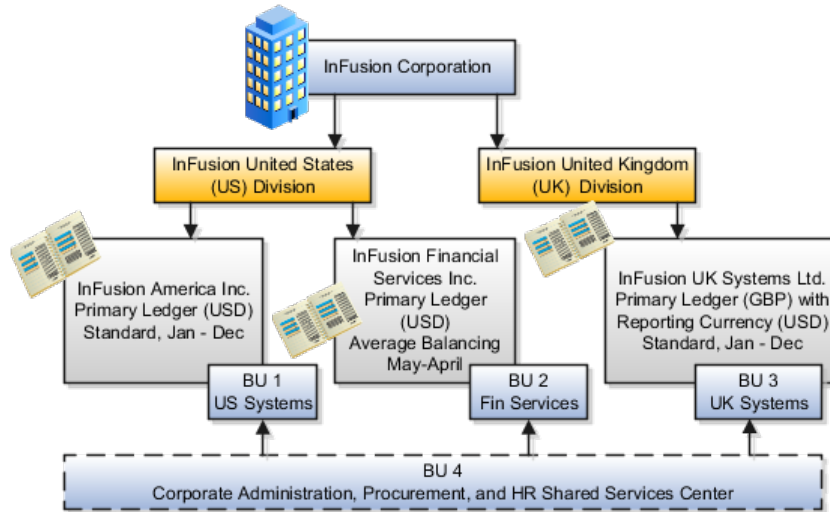
Service Provider Model: Explained

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

Procurement Example

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

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



Define Business Units for Customer Data Management: Specify Customer Contract Management Business Function Properties

Customer Contracts Business Unit Setup: Explained

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

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

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

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

Enabling Related Customer Accounts

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

Managing Currency Conversion Options

If your organization plans to transact project-related business in multiple currencies, then select the multicurrency option. This allows a contract author

to override a contract's currency, which defaults from the ledger currency of the business unit. It also enables the contract author to specify currency conversion attributes to use when converting from the bill transaction currency to the contract currency and from the invoice currency to the ledger currency.

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

In the Invoice Currency to Ledger Currency region:

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

Managing Project Billing Options

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

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

Set these options, which apply to all contracts:

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

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

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

Set this option only for customer billing:

- Indicate if you want contract authors to manually enter the Receivables transaction type on the customer contracts they create.

Contract Terms Library Business Unit Setup: Explained

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

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

- Enable clause and template adoption.
- Set the clause numbering method.
- Enable the Contract Expert feature.
- Specify the layout for printed clauses and contract deviation reports.

Enabling Clause Adoption

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

- Specify a global business unit

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

- Enable automatic adoption

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

- Specify the administrator who approves clauses available for adoption

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

Setting Clause Numbering Options

You can set up automatic clause numbering for the clauses in the business unit by selecting Automatic in the **Clause Numbering** field and entering a Document Sequence Category you previously set up in the **Clause Sequence Category**

field. If clause numbering is manual, contract terms library administrators must enter unique clause numbers each time they create a clause.

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

Enabling Contract Expert

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

Specifying the Printed Clause and Deviations Report Layouts

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

- The printed contract terms

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

- The contract deviations report

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

Define Business Units for Customer Data Management: Specify Supplier Contract Management Business Function Properties

Supplier Contracts Business Unit Setup: Explained

Using the **Specify Supplier Contract Management Business Function Properties** task, available by selecting Setup and Maintenance from the Tools menu and searching on the task name, you can specify a variety of business function settings for supplier contracts in a specific business unit.

The selections you make for these business functions impact how the Contract Terms Library behaves during supplier contract authoring.

Managing Contract Terms Library Setup Options

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

Contract Terms Library Business Unit Setup: Explained

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

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

- Enable clause and template adoption.
- Set the clause numbering method.
- Enable the Contract Expert feature.
- Specify the layout for printed clauses and contract deviation reports.

Enabling Clause Adoption

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

- Specify a global business unit

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

- Enable automatic adoption

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

- Specify the administrator who approves clauses available for adoption

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

Setting Clause Numbering Options

You can set up automatic clause numbering for the clauses in the business unit by selecting Automatic in the **Clause Numbering** field and entering a Document Sequence Category you previously set up in the **Clause Sequence Category** field. If clause numbering is manual, contract terms library administrators must enter unique clause numbers each time they create a clause.

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

Enabling Contract Expert

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

Specifying the Printed Clause and Deviations Report Layouts

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

- The printed contract terms

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

- The contract deviations report

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

Define Business Units for Customer Data Management: Assign Business Unit Business Function

Business Functions: Explained

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

Business Functions

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

- Billing and revenue management
- Collections management
- Customer contract management
- Customer payments
- Expense management
- Incentive compensation
- Marketing
- Materials management
- Inventory management
- Order fulfillment orchestration
- Payables invoicing
- Payables payments

- Procurement
- Procurement contract management
- Project accounting
- Receiving
- Requisitioning
- Sales

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

Note

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

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

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

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

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

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

- Requisitioning

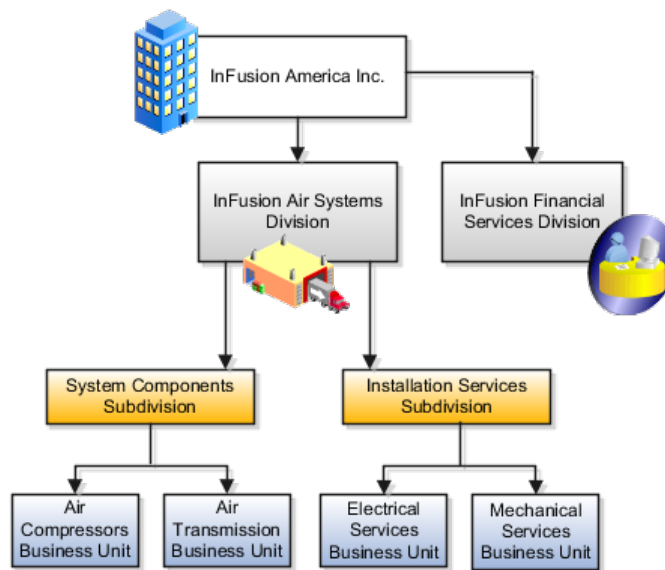
Business Unit Hierarchy: Example

For example, your InFusion America Company provides:

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

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

- System Components by products: Air Compressors and Air Transmission
- Installation Services by services: Electrical and Mechanical



Oracle Fusion applications facilitates independent balance sheet rollups for legal and management reporting by offering up to three balancing segments. Hierarchies created using the management segment can provide the divisional results. For example, it is possible to define management segment values to correspond to business units, and arrange them in a hierarchy where the higher nodes correspond to divisions and subdivisions, as in the InFusion US Division example above.

Define Business Units for Customer Data Management: Manage Business Units

Business Units: Explained

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. A business unit can process transactions on behalf of many legal entities. Normally, it will have a manager, strategic objectives, a level of autonomy, and responsibility for its profit

and loss. Roll business units up into divisions if you structure your chart of accounts with this type of hierarchy. In Oracle Fusion Applications, you assign your business units to one primary ledger. For example, if a business unit is processing payables invoices they will need to post to a particular ledger. This assignment is mandatory for your business units with business functions that produce financial transactions.

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

The Oracle Fusion Applications business unit model:

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

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

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

In summary, use business units in the following ways:

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

Brief Overview of Business Unit Security

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

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

Define Workforce Structures for CRM: Manage Locations

Locations: Explained

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

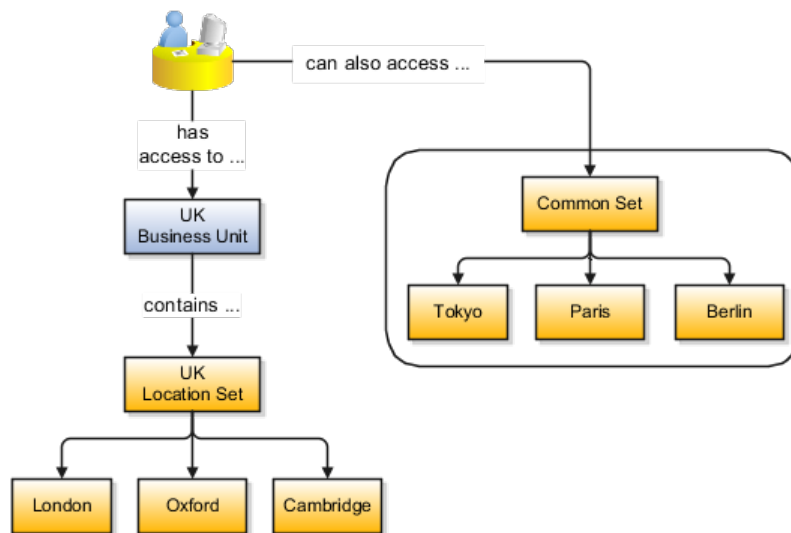
The locations that you create exist as separate structures that you can use for reporting purposes, and also in rules that determine employee eligibility for various types of compensation and benefits. You enter information about a location only once. Subsequently, when you set up other workforce structures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

Location Sets

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

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

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



Creating Multiple Locations Simultaneously

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

Define Workforce Structures for CRM: FAQs for Manage Locations

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

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

How can I associate a location with an inventory organization?

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

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

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

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

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

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

What happens if I inactivate a location?

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

Define Workforce Structures for CRM: Manage Divisions

Division: Explained

Managing multiple businesses requires that you segregate them by their strategic objectives and measure their results. Responsibility to reach objectives can be delegated along the management structure. Although related to your legal structure, the business organizational hierarchies do not need to reflect directly the legal structure of the enterprise. The management entities and structure can include divisions and subdivisions, lines of business, and other strategic business units, and include their own revenue and cost centers. These organizations can

be included in many alternative hierarchies and used for reporting, as long as they have representation in the chart of accounts.

Divisions

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

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

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

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

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

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

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

Adding a New Division After Acquiring a Company: Example

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

Scenario

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

Details of the Acquired Company

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

Analysis

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

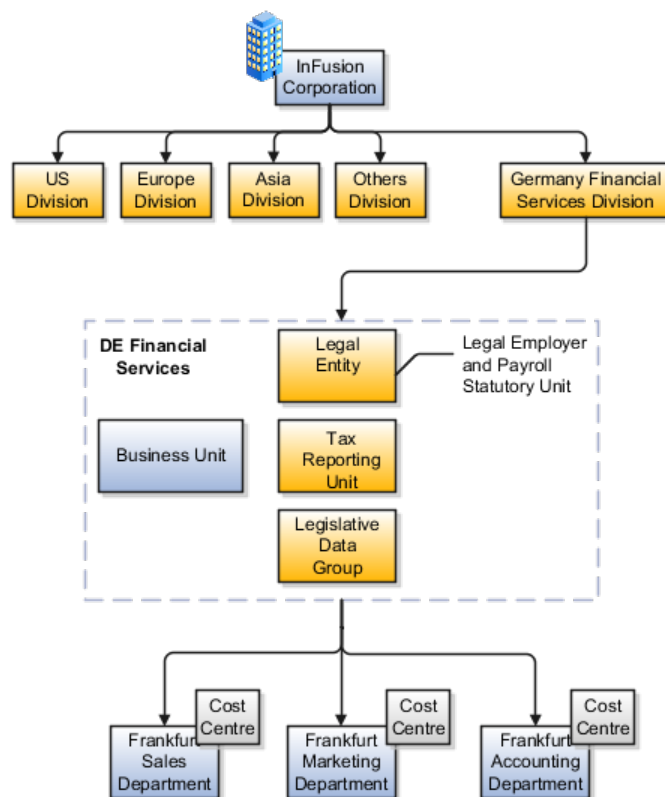
Decision to Consider	In This Example
Create location?	The financial services company is based in Frankfurt as are the departments, so you need to create only one location.
Create separate division?	Yes. Although the new division will exist within the current enterprise structure, you want to keep the financial services company as a separate line of business. Creating a separate division means you can manage the costs and reporting separately from the InFusion Corporation. It also means you do not have to modify any existing organizations in the enterprise setup.
Create business unit?	Yes. The financial services business requires you to create several jobs that do not exist in your high tech business. You can segregate the jobs that are specific to financial services in a new business unit.
How many departments?	The financial services company currently has three departments for sales, accounting, and marketing. As you have no plans to downsize or change the company, you can create three departments to reflect this structure.
How many cost centers?	Although you can have more than one cost center tracking the costs of a department, you decide to create one cost center for each department to track costs.
How many legal entities?	<p>Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, and income, pay transaction taxes, or perform intercompany trading. In this case, you need only one legal entity.</p> <p>You must define the legal entity as a legal employer and payroll statutory unit. As the new division operates in Germany only, you can configure the legal entity to suit Germany legal and statutory requirements.</p> <hr/> <p>Note</p> <p>When you identify the legal entity as a payroll statutory unit, the application transfers the legal reporting unit that is associated with that legal entity to Oracle Fusion HCM as a tax reporting unit.</p>

Create legislative data group?	Yes. Because you currently do not employ or pay people in Germany, you must create one legislative data group to run payroll for the workers in Germany.
--------------------------------	--

Based on the analysis, you must create the following:

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

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



Define Workforce Structures for CRM: Manage Departments

Cost Centers and Departments: Explained

A cost center represents the smallest segment of an organization for which costs are collected and reported. A department is an organization with one or more

operational objectives or responsibilities that exist independently of its manager and has one or more workers assigned to it.

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

- Cost centers
- Departments

Cost Centers

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

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

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

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

Departments

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

A manager of a department is typically responsible for:

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

Note

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

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

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

Department Classifications: Points to Consider

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

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

Project Organization

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

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

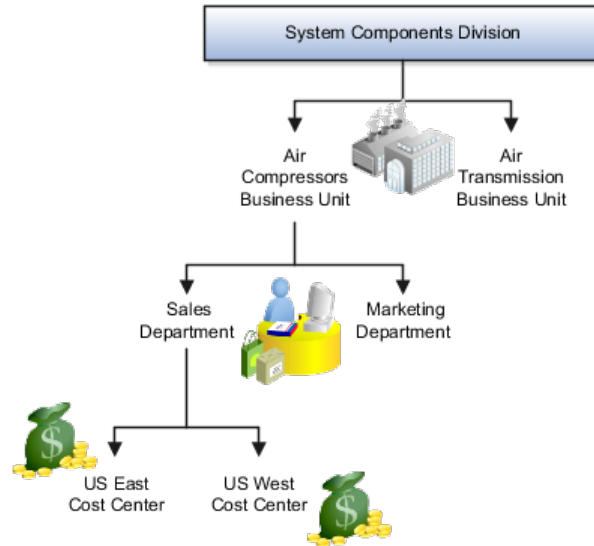
Sales and Marketing Organization

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

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

The following figure illustrates a management hierarchy, in which the System Components Division tracks its expenses in two cost centers, Air Compressors and Air Transmission. At the department level, two organizations with a

classifications of Department are defined, the Marketing Department and Sales Department. These two departments can be also identified as a Resource Organizations, which will allow assigning resources, such as sales people, and other CRM specific information to them. Each department is represented in the chart of accounts by more than one cost center, allowing for granular as well as hierarchical reporting.



Cost Organization

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

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

Define Workforce Structures for CRM: FAQs for Manage Job Families

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

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

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

Define Workforce Structures for CRM: Manage Jobs

Jobs: Explained

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

Basic Details

Basic details for a job include an effective start date, a job set, a name, and a code.

A job code must be unique within a set. Therefore, you can create a job with the code DEV01 in the US set and another job with the same code in the UK set. However, if you create a job with the code DEV01 in the Common set, then you cannot create a job with the same code in any other set.

Benchmark Information

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

Progression Information

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

Jobs and Grades

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

Evaluation Criteria

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

Uploading Jobs Using a Spreadsheet

If you have a list of jobs already defined for your enterprise, you can upload them from a spreadsheet. To use this option, you first download a spreadsheet

template, then add your job information to the spreadsheet, and then upload directly to your enterprise configuration. You can upload the spreadsheet multiple times to accommodate revisions.

Jobs: Example

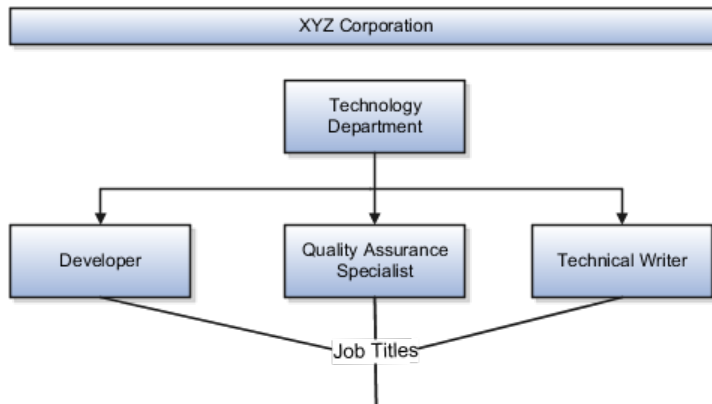
Jobs are typically used without positions by service industries where flexibility and organizational change are key features.

Software Industry

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

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

This figure illustrates the software industry job setup.



Common Applications Configuration: Define Security for CRM

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 (Oracle Fusion Applications Edition).
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 (Oracle Fusion Applications Edition).

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.

See: Common Applications Configuration: Define Security for Human Capital Management

- For a complete description of the Oracle Fusion Applications security reference implementation, see the Oracle Fusion Applications Security Reference Manuals for each offering.

See: Oracle Fusion Applications Common Security Reference Manual

- 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 (Oracle Fusion Applications Edition).

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 Customer Relationship Management (CRM) Self Service, respectively, and associate them with users.

Before Importing Users

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the 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 CRM.

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

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

Import Users

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

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the 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, CRM or Oracle Fusion Supplier Portal.

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

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

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

Manage Duties

A person with a job role must be able to perform certain duties. In the Oracle Fusion Applications security reference implementation, enterprise roles inherit

duties through a role hierarchy. Each duty corresponds to a duty role. Duty roles specify the duties performed within applications and define the function and data access granted to the enterprise roles that inherit the duty roles.

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

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.

In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

Tip

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

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

Note

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

Manage Application Access Controls

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

Note

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

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

Security Tasks and Oracle Fusion Applications: How They Fit Together

The major security tasks and their order within the context of an overall Oracle Fusion Applications implementation extend from security setup through production deployment audits.

The Oracle Fusion business process model (BPM) provides a sequence of security implementation tasks that includes the following.

- Security setup (Define Common Applications Configuration activity)
 - Define Implementation Users task group (optional)
 - Create Implementation Users task
 - Create Data Role for Implementation Users task
 - 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 Encryption Keys task
 - Manage Segment Security task
 - Manage Data Access Sets task
 - Define Security Profiles task group
 - Auditing tasks
 - Manage Security Audit, Compliance and Reporting activity
 - Manage Application Access Controls task

Note

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

Required Roles

The 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.

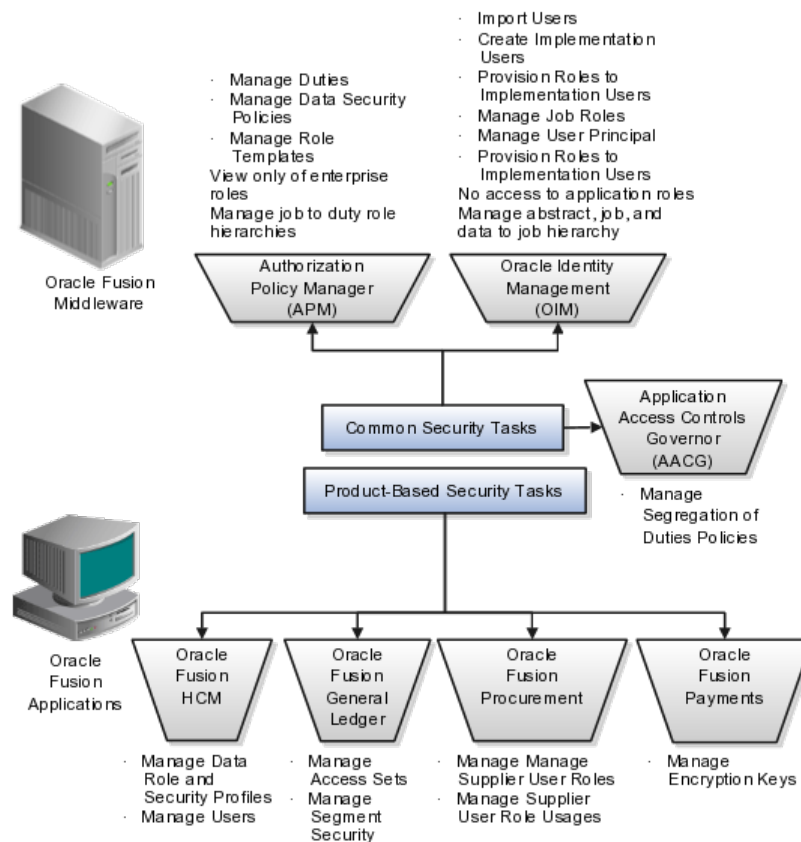
Important

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.

Example Task	Oracle BPM Task	Supporting Tools	Details
View duty roles inherited by a job role	Manage Duties	<ul style="list-style-type: none"> Authorization Policy Manager (APM) 	Each logical partition or pillar contains a collection of application roles, and function and data security policies.
View entitlement or policies carried by a job role	Manage Duties	<ul style="list-style-type: none"> APM 	In LDAP, the policy store stores application roles and the identity store stores enterprise roles.
Add a job role to a role hierarchy	Manage Job Roles	<ul style="list-style-type: none"> Oracle Identity Management (OIM) 	The identity store in LDAP stores enterprise roles.
Add a duty role to a role hierarchy	Manage Duties	<ul style="list-style-type: none"> APM 	LDAP stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions.
Create a hierarchy of enterprise (abstract, job, data) roles	Manage Job Roles	<ul style="list-style-type: none"> OIM 	
Create a hierarchy of (application) duty roles	Manage Duties	<ul style="list-style-type: none"> APM 	
Create a new job role	Manage Job Roles	<ul style="list-style-type: none"> OIM 	The identity store in LDAP stores enterprise roles.
Change duty roles inherited by a job or abstract role	Manage Duties	<ul style="list-style-type: none"> APM 	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.
Create a new duty role	Manage Duties	<ul style="list-style-type: none"> APM 	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.

View Segregation of Duties (SOD) policies respected by a duty role	Manage Application Access Controls	<ul style="list-style-type: none"> Application Access Controls Governor (AACG) in Governance, Risk, and Compliance Controls (GRCC) 	The Security Reference Manuals (SRM) document the segregation of duties (SOD) policies respected within each job role
View SOD policy violations carried by the duty roles inherited by a job role	Manage Application Access Controls	<ul style="list-style-type: none"> AACG in GRCC 	The Security Reference Manuals (SRM) document the SOD policies respected within each job role
View SOD policy violations	Manage Segregation of Duties Policies	<ul style="list-style-type: none"> AACG in GRC 	The SRM documents the SOD conflicts for each job role
View the data security policies carried by a job, abstract, and data roles	Manage Data Security Policies	<ul style="list-style-type: none"> APM 	<p>Oracle Fusion Data Security stores data security policies in the policy store.</p> <p>Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)</p>
Create and update HCM security profiles	Manage Data Role and Security Profiles	<ul style="list-style-type: none"> Oracle Fusion HCM 	This task does not include assigning data roles to the users, which is supported by user provisioning tasks.
Create (generate) a data role	<ol style="list-style-type: none"> Manage Role Templates Manage Data Roles and Security Profiles 	<ul style="list-style-type: none"> APM Oracle Fusion HCM 	<p>Data roles are generated automatically based on data role templates and enterprise setup. Changes to data role templates generate new or changed data roles.</p> <p>Create data roles in HCM using the Manage Data Roles and Security Profiles task.</p>
Create a new data security policy (not through generated data roles based on data role templates or HCM security profiles)	Manage Data Security Policies	<ul style="list-style-type: none"> APM 	Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)
View data role templates defined by a product	Manage Role Templates	<ul style="list-style-type: none"> APM 	
Create or edit an existing data role template	Manage Role Templates	<ul style="list-style-type: none"> APM 	

Secure common objects such as attachment categories or profile options	Manage Data Security Policies	<ul style="list-style-type: none"> • APM 	Data security can also be defined in application pages provided by Oracle Middleware Extensions for Applications (FND)
View, create, update encryption keys used to secure attributes of personally identifiable information	Manage Encryption Keys	<ul style="list-style-type: none"> • Oracle Fusion Payments 	
View, create, update Data Access Sets used to secure Ledgers and Ledger Sets	Manage Data Access Sets	<ul style="list-style-type: none"> • Oracle Fusion General Ledger 	
View, create, update accounting flexfield segment security rules	Manage Security Segments	<ul style="list-style-type: none"> • Oracle Fusion General Ledger 	
View or update the set of job roles that can be provisioned to supplier users	Manage Supplier User Role	<ul style="list-style-type: none"> • Supplier Portal • Sourcing 	These tools are in the Oracle Fusion Procurement product family
Determine the supplier job roles that the supplier self service administrator can provision to supplier users	Manage Supplier User Role Usages	<ul style="list-style-type: none"> • Supplier Portal • Sourcing 	These tools are in the Oracle Fusion Procurement product family
Set default supplier job roles based on the set of supplier roles that are defined by performing the Manage Supplier User Roles task	Manage Supplier User Role Usages	<ul style="list-style-type: none"> • Supplier Portal • Sourcing 	These tools are in the Oracle Fusion Procurement product family
Create a new implementation user	Create Implementation Users	<ul style="list-style-type: none"> • OIM 	
Import legacy users	<ul style="list-style-type: none"> • Import Worker Users • Import Partner Users 	<ul style="list-style-type: none"> • OIM 	
Create a new user	Manage Users	<ul style="list-style-type: none"> • HCM 	<p>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.</p> <p>Creating a new user automatically triggers role provisioning requests based on role provisioning rules.</p>

Provision roles to a user	<ol style="list-style-type: none"> 1. Provision Roles to Implementation Users 2. Manage Users 	<ul style="list-style-type: none"> • OIM • Oracle Fusion HCM • Oracle Fusion CRM • Oracle Fusion Suppliers 	<p>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.</p> <p>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.</p> <p>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.</p>
View the job, abstract, and data roles provisioned to a user	<ol style="list-style-type: none"> 1. Manage Users 2. Manage User Principal 3. Provision Roles to Implementation Users 	<ul style="list-style-type: none"> • Human Capital Management (HCM) • OIM 	<p>LDAP stores users, roles and provisioning information.</p> <p>The Manage User Principal and Provision Roles to Implementation Users tasks are not needed once implementation is complete.</p>
Revoke role from user.	Manage Users	<ul style="list-style-type: none"> • HCM 	You can revoke roles from various Human Resources task flows, the HCM Manage Users task and OIM. User termination includes role revocation.
Approve role provisioning or user account request.	Approve User and Role Provisioning Requests	<ul style="list-style-type: none"> • OIM 	
View audit logs	Not applicable	<ul style="list-style-type: none"> • Oracle Enterprise Manager 	Viewing audit logs is a Oracle Fusion Middleware function and not represented by an Oracle Fusion Applications BPM task.

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

There may be more than one navigation path to the graphical user interface in which the task is performed. You can access most security tasks by starting in the Setup and Maintenance Overview page and searching for security tasks and task lists.

Define Data Security

Data Security: Explained

By default, users are denied access to all data.

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

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

You secure data by provisioning roles that provide the necessary access. Enterprise roles provide access to data through data security policies defined for the inherited application roles.

When setting up the enterprise with structures such as business units, data roles are automatically generated that inherit job roles based on data role templates. Data roles also can be generated based on HCM security profiles. Data role templates and HCM security profiles enable defining the instance sets specified in data security policies.

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

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

Data is secured by the following means.

Data security feature	Does what?
Data security policy	Grants access to roles by means of entitlement
Role	Applies data security policies with conditions to users through role provisioning.

Data role template	Defines the data roles generated based on enterprise setup of data dimensions such as business unit.
HCM security profile	Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code
Masking	Hides private data on non-production database instances
Encryption	Scrambles data to prevent users without decryption authorization from reading secured data

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.

Who	can do	what	on which set of data
Accounts payable managers	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.

Oracle Fusion Payroll uses HCM security profiles to secure project organizations. Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

Masking and Encryption

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

Defining Data Security After Enterprise Setup: Points to Consider

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

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

- Manage Data Access Sets
- Manage Segment Security
- Manage Role Templates
- Manage Data Security Policies
- Manage Encryption Keys

These tasks address data security administration. For information on using the user interface pages for setting up and managing data security, see the Oracle

Note

The Manage Data Role and Security Profiles task, and all other HCM security profile setup tasks are documented in Human Capital Management (HCM) documentation.

Manage Data Access Sets

Data access sets define a set of access privileges to one or more ledgers or ledger sets.

The information on ledgers that are attached to data access sets are secured by function security. Users must have access to the segment values associated with the data access sets to access the corresponding GL account.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Data Access Administration Duty role, which is entitled to manage data access sets (the entitlement is Define General Ledger Data Access Set). This entitlement provides the access necessary to perform the Manage Data Access Sets task in General Ledger.

Manage Segment Security

Balancing or management segment values can secure data within a ledger.

Segment values are stored in GL_ACCESS_SET_ASSIGNMENTS and secured by restrictions, such as Exclude, on parameters that control the set of values that a user can use during data entry.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Application Key Flexfield Administration Duty role, which is entitled to manage application key flexfields (the entitlement is Manage Application Key Flexfield). This entitlement provides the access necessary to perform the Manage Segment Security task in General Ledger.

Manage Role Templates

Data role templates automatically create or update data roles based on dimensions such as business unit. As an enterprise expands, data role templates trigger replication of roles for added dimensions. For example, when creating a new business unit, a data role template generates a new Accounts Payables Manager data role based on the Financials Common Module Template for Business Unit Security data role template.

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

Manage Data Security Policies

Data security grants provisioned to roles are data security policies. The security reference implementation provides a comprehensive set of predefined data security policies and predetermined data security policies based on data role templates.

Data security policies are available for review in Authorization Policy Manager (APM). Data security policies are implemented by grants stored in Oracle Fusion Data Security (FND_GRANTS).

Data security policies secure the database resources of an enterprise. Database resources are predefined applications data objects and should not be changed. However, for cases where custom database resources must be secured objects, the IT security manager is entitled to manage database resources and create new data security policies.

Warning

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

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

Manage Encryption Keys

Create or edit encryption keys held in Oracle Wallet to secure Personally Identifiable Information (PII) attributes. This task is only available when Payments is implemented.

In the security reference implementation, the IT Security Manager job role hierarchy includes the Payments Data Security Administration Duty role, which is entitled to manage encryption keys that secure PII (the entitlement is Manage Wallet). This entitlement provides the access necessary to perform the Manage Encryptions Keys task in Payments.

Data Security in the Security Reference Implementation: Explained

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

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

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

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

Data Security Policies in the Security Reference Implementation

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

Tip

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

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

Warning

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

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

HCM Security Profiles in the Security Reference Implementation

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

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

Data Roles

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

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

HCM provides a mechanism for generating HCM related data roles.

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.

Data security feature	Does what?
Data security policy	Grants access to roles by means of entitlement
Role	Applies data security policies with conditions to users through role provisioning.
Data role template	Defines the data roles generated based on enterprise setup of data dimensions such as business unit.

HCM security profile	Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code
Masking	Hides private data on non-production database instances
Encryption	Scrambles data to prevent users without decryption authorization from reading secured data

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.

Who	can do	what	on which set of data
Accounts payable managers	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.

Oracle Fusion Payroll uses HCM security profiles to secure project organizations. Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

Masking and Encryption

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

Securing Data Access: Points to Consider

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

Method of securing data	Reason	Example
Data roles apply explicit data security policies on job and abstract roles	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.	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.
Data security policies	Define data access for application roles and provide inheriting job and abstract roles with implicit data security	Projects

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.

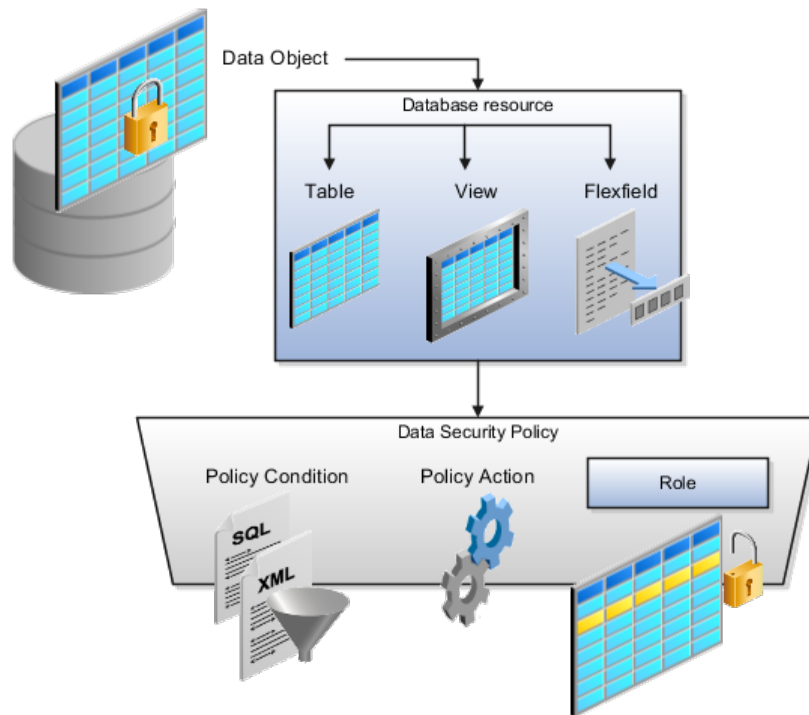
Manage Data Security Policies

Database Resources and Data Security Policies: How They Work Together

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

The database resource defines an 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 a role access to that database resource based on the policy's action and condition.



Database Resources

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

- Name providing a means of identifying the database resource

- Data object to which the database resource points

Data Security Policies

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

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

Note

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

Actions

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

Conditions

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

Tip

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

Manage Role Templates

Data Role Templates: Explained

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

Note

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

The following attributes define a data role template.

- Template name

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

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

Note

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

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

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

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

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

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

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

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

Making Changes To Data Role Templates

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

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

Making Changes to Dimension Values

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

If you add or remove a dimension value from your enterprise that is used to generate data roles, regenerating the set of data roles adds or removes the data roles for those dimension values. If your enterprise has scheduled regeneration as an Oracle Enterprise Scheduler Services process, the changes are made automatically.

For information on working with data role templates, see the Oracle Fusion Middleware Administrator's Guide for Authorization Policy Manager (Oracle Fusion Applications Edition).

Manage Data Role and Security Profiles

HCM Data Roles: Explained

HCM data roles, like all Oracle Fusion Applications data roles, define data security policies: they enable users to perform a set of tasks, using identified menus, menu items, and pages in application user interfaces, on a specified set of data within those user interfaces. Because data roles are specific to the enterprise, no predefined HCM data roles exist.

How HCM Data Roles Differ from Other Data Roles

HCM data roles differ from other data roles in the following ways:

- You create and maintain HCM data roles outside Oracle Identity Management (OIM) and the Oracle Fusion Middleware Authorization Policy Manager (APM), and they are not based on data role templates.

Although HCM data roles are visible in the Oracle Fusion Middleware APM, they must not be maintained there.

- A single HCM data role can enable access to data of multiple types.

You identify the data that users can access in HCM security profiles. You can create security profiles for the person, organization, position, country, legislative data group (LDG), document type, payroll, payroll flow, and workforce business process objects.

Selecting the Job Role

Each HCM data role is associated with a single job role, which you select from the list of enterprise roles. The HCM securing objects that the selected role needs to access are identified automatically, and the appropriate types of security profile are displayed. For example, if you select the job role human resource analyst, users with that job role need to access managed person, public person, organization, position, LDG, and document type data; therefore, security profiles for those object types must be included in the HCM data role. The security profile types that appear in the HCM data role vary according to the data requirements of the selected job role.

If you select a job role that requires no access to HCM data secured by security profiles, you cannot create an HCM data role.

Note

If you create custom job roles in OIM, you must add them to a locally defined role category that ends with "Job Roles"; otherwise, they do not appear in the list of job roles when you create an HCM data role. Do not add custom job roles to the predefined role category HCM - Job Roles.

Creating or Selecting the Security Profiles

You can either create new security profiles or use existing security profiles. For each object type, you can include only one security profile in an HCM data role.

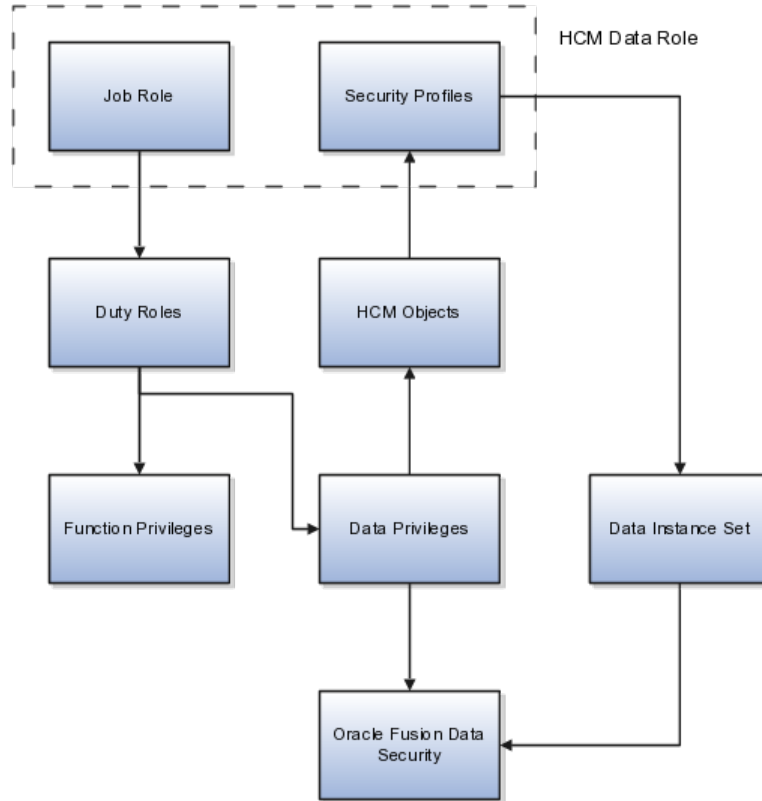
Users with Multiple HCM Data Roles

When users have multiple HCM data roles, the data security policies arising from each role remain separate. For example, being able to promote or terminate workers in the purchasing department in one HCM data role and view contact details of all workers in the sales department in another HCM data role does not enable a user to promote or terminate workers in the sales department.

Components of the HCM Data Role

The following figure summarizes how the components of the HCM data role contribute to Oracle Fusion Data Security for the data role. Oracle Fusion Data Security comprises the data security policies for data roles that are generated automatically when data roles are created.

The job role that you select in the HCM data role inherits multiple duty roles. Each duty role has one or more function privileges and related data privileges, from which the relevant HCM objects are identified. The specific instances of the objects required by this HCM data role are identified in security profiles and stored in a data instance set. Data security policy data is created automatically in Oracle Fusion Data Security when you create the data role.



For example, the human resource specialist job role inherits the employee hire and worker promotion duty roles, among many others. The inherited duty roles provide both function privileges, such as Hire Employee, Rehire Employee, and Promote Workers, and data privileges to HCM objects, such as person and assignment. The specific instances of those objects required by this HCM data role, such as people with assignments in a specified legal employer and department, are identified in security profiles.

HCM Security Profiles: Explained

A security profile defines the criteria that identify instances of a human capital management (HCM) object. For example, a person security profile defines the criteria that identify one or more person records, and a position security profile defines the criteria that identify one or more positions. When you include a security profile in an HCM data role and provision the data role to a user, that user can access the data instances identified in the security profile. The type of access available to the user (for example whether the user can edit or simply view the data) depends on the job role identified in the HCM data role.

HCM Object Types

You can create security profiles for the following HCM object types:

- Person
 - Managed person
 - Public person

- Organization
- Position
- Legislative data group (LDG)
- Country
- Document type
- Payroll
- Payroll flow
- Workforce business process

All security profile definitions for these HCM objects are eventually visible in the Oracle Fusion Middleware Authorization Policy Manager (APM). The name of the security profile's data instance set in the Oracle Fusion Middleware APM is derived from the name of the security profile and the relevant object type. For example, if the security profile name is Manager Hierarchy, then the data instance set for the object PER_ALL_PEOPLE_F is HCM:PER:PER_ALL_PEOPLE_F:Manager Hierarchy.

You must use the Oracle Fusion Human Capital Management interfaces, which are designed for ease of use and access, to create and maintain security profiles; do not use the Oracle Fusion Middleware APM to maintain security profiles for these HCM objects.

Security Criteria in HCM Security Profiles

In any HCM security profile, you specify the criteria that identify data instances of the relevant type. For example, in an organization security profile, you can identify organizations by organization hierarchy, by organization classification, or by listing organizations to include in or exclude from the security profile. All of the criteria in an HCM security profile apply when the data instance set is defined; for example, if you identify organizations by both organization hierarchy and organization classification, then both sets of criteria apply, and only those organizations that satisfy all criteria belong to the data instance set.

Predefined HCM Security Profiles

The following HCM security profiles are predefined:

Security Profile Name	HCM Security Profile Type	Description
View All People	Person	Identifies all person records in the enterprise
View Own Record	Person	Identifies the signed-on user's own person record and the person records of that user's contacts
View Manager Hierarchy	Person	Identifies the signed-on user's line manager hierarchy
View All Workers	Person	Identifies the person records of all people who have a work relationship in the enterprise
View All Organizations	Organization	Identifies all organizations in the enterprise

View All Positions	Position	Identifies all positions in the enterprise
View All Legislative Data Groups	LDG	Identifies all LDGs in the enterprise
View All Countries	Country	Identifies all countries in the FND_TERRITORIES table
View All Document Types	Document Type	Identifies all document types in the enterprise
View All Payrolls	Payroll	Identifies all payrolls in the enterprise
View All Flows	Payroll Flow	Identifies all payroll flows in the enterprise
View All Workforce Business Processes	Workforce Business Process	Identifies all registered workforce business processes in the enterprise

You can include the predefined security profiles in any HCM data role, but you cannot edit them. Note also that the **View all** option is disabled in any security profile that you create; this restriction exists because predefined security profiles exist for this requirement.

Creating Security Profiles

You can create security profiles either individually or as part of the process of creating an HCM data role. If you have standard requirements, it may be more efficient to create the security profiles individually and include them in appropriate HCM data roles.

Reusability and Inheritance of Security Profiles

Regardless of how you create them, all security profiles are reusable; they do not belong to particular HCM data roles, and you can include them in any HCM data role for which they define an appropriate data instance set.

You can include security profiles in other security profiles. For example, you can include an organization security profile:

- In a person security profile, to secure person records by department, business unit, or legal employer
- In a position security profile, to secure positions by department or business unit

Therefore, one security profile can inherit the data instance set defined by another.

Assigning Security Profiles to Abstract Roles: Explained

Abstract roles define a worker's role in the enterprise independently of the job that the worker is hired to do.

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

- Line manager

- Employee
- Contingent worker

Enabling Data Access for Abstract Roles

Typically, you create role mappings during implementation to provision abstract roles automatically to eligible workers. Although users with these roles may be able to sign in to Oracle Fusion Applications and navigate to tasks of interest, they have no automatic access to data. For example, employees can navigate to the Person Gallery but cannot view portraits or see lists of person names in product interfaces, and line managers can navigate to the Manager Resources Dashboard but can see no data for their organizations. To enable users with abstract roles to access relevant HCM data, you must assign security profiles to those abstract roles.

Predefined Security Profiles to Assign to Abstract Roles

To enable users with abstract roles to access relevant data, you assign the following predefined security profiles directly to the employee, contingent worker, and line manager abstract roles.

Security Profile Type	Employee	Contingent Worker	Line Manager
Person	View Own Record	View Own Record	View Manager Hierarchy
Public person	View All Workers	View All Workers	View All Workers
Organization	View All Organizations	View All Organizations	View All Organizations
Position	View All Positions	View All Positions	View All Positions
Legislative data group	View All Legislative Data Groups	View All Legislative Data Groups	View All Legislative Data Groups
Country	View All Countries	View All Countries	View All Countries
Document type	View All Document Types	View All Document Types	View All Document Types
Payroll Flow	Not applicable	Not applicable	View All Flows
Workforce Business Process	Not applicable	Not applicable	View All Workforce Business Processes

After implementation, you may want to change aspects of this data access. For example, you may want to create your own security profiles and assign those directly to abstract roles; however, you must remember that such changes apply to all users who have the abstract role.

HCM Data Roles

Users who have abstract roles are likely to gain additional data access by means of HCM data roles that you define for their job roles. For example, you may create an HCM data role for human resource specialists to enable them to access the person records of all workers in a legal employer. Such data access is in addition to any data access provided by abstract roles.

Assigning Security Profiles to Abstract Roles: Worked Example

This example shows how to assign predefined security profiles to the employee, contingent worker, and line manager abstract roles.

Searching for the Employee Abstract Role

1. On the All Tasks tab of the Overview page of the Setup and Maintenance work area, search for the task Manage Data Role and Security Profiles.
2. In the Search Results region, click **Go to Task**.
3. On the Manage Data Roles and Security Profiles page, enter the abstract-role name Employee in the **Role** field. Click **Search**.
4. In the Search Results region, highlight the entry for the predefined Employee role and click **Assign**.

Assigning Security Profiles to the Employee Abstract Role

1. On the Assign Data Role: Security Criteria page, select the security-profile values shown in the following table. These are the security profiles that are typically assigned to the employee role. You may see a subset of these security profiles, depending on the combination of product offerings that you are implementing.

Field	Value
Organization Security Profile	View All Organizations
Position Security Profile	View All Positions
Country Security Profile	View All Countries
LDG Security Profile	View All Legislative Data Groups
Person Security Profile (Person section)	View Own Record
Person Security Profile (Public Person section)	View All Workers
Document Type Security Profile	View All Document Types

2. Click **Review**.
3. On the Assign Data Role: Review page, click **Submit**.
4. On the Manage Data Roles and Security Profiles page, search again for the predefined Employee role.
5. In the Search Results region, confirm that a green check mark appears in the **Security Profiles** column for the Employee role. The check mark confirms that security profiles are assigned to the role.

Repeat the steps in Searching for the Employee Abstract Role and Assigning Security Profiles to the Employee Abstract Role for the predefined Contingent Worker role.

Searching for the Line Manager Abstract Role

1. On the Manage Data Roles and Security Profiles page, enter the abstract-role name Line Manager in the **Role** field. Click **Search**.
2. In the Search Results region, highlight the entry for the predefined Line Manager role and click **Assign**.

Assigning Security Profiles to the Line Manager Abstract Role

1. On the Assign Data Role: Security Criteria page, select the security-profile values shown in the following table. These are the security profiles that are typically assigned to the line manager role. You may see a subset of these security profiles, depending on the combination of product offerings that you are implementing.

Field	Value
Organization Security Profile	View All Organizations
Position Security Profile	View All Positions
LDG Security Profile	View All Legislative Data Groups
Person Security Profile (Person section)	View Manager Hierarchy
Person Security Profile (Public Person section)	View All Workers
Document Type Security Profile	View All Document Types
Payroll Flow	View All Flows
Workforce Business Process	View All Workforce Business Processes

2. Click **Review**.
3. On the Assign Data Role: Review page, click **Submit**
4. On the Manage Data Roles and Security Profiles page, search again for the predefined Line Manager role.
5. In the search results, confirm that a green check mark appears in the **Security Profiles** column for the Line Manager role. The check mark confirms that security profiles are assigned to the role.

Define Users

Securing Identities and Users: Points To Consider

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

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

Roles are part of a user's identity and define the user's purpose and responsibilities.

Within identity management, resources define what a user can and does do. In an enterprise, this typically translates into what resources a user has access to, what privileges they have on that resource, and what they have been doing on that resource. Resources can be application accounts or physical devices such as

laptops or access cards. The enterprise owns the resources, secures them, and manages access to the resources by managing the user's identity and access.

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

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

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

Stores

Various locations store identity and user data.

Identity data consists of the following.

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

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

Importing Users

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

Your tasks may include the following.

- Create users in bulk
- Update specific attributes for all users, such as postal code
- Link users to HR or Trading Community Model persons
- Monitor progress of the import process
- Correct errors & re-import
- Export users in bulk

- Import and export users using a standard plain text data interchange format like Lightweight Data Interchange Format (LDIF)

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

Provisioning Events

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

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

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

Notifications and Audits

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

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

User ID and GUID attributes are available in Oracle Fusion Applications session information for retrieving authenticated user and identity data.

End user auditing data is stored in database WHO columns and used for the following activities.

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

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

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

Delegated Administration

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

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

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

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

Credentials

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

Applications themselves must be credentialed to access one another.

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

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

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

Manage HCM Role Provisioning Rules

Role Provisioning and Deprovisioning: Explained

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

Role Provisioning Methods

Roles can be provisioned to users:

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

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

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

Role Types

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

Automatic Role Provisioning

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

Role Deprovisioning

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

Automatically provisioned roles can be deprovisioned manually at any time.

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

Changes to Assignment Managers

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

Roles at Termination

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

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

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

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

Date-Effective Changes to Assignments

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

Role Mappings: Explained

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

The role mapping can support:

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

Automatic Provisioning of Roles to Users

A role is provisioned to a user automatically if:

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

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

Attribute	Value
Department	Finance Department
Job	Sales Manager
Assignment Status	Active

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.

Attribute	Value
Manager with Reports	Yes
Assignment Status	Active

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.

Attribute	Value
Department	ABC Department
System Person Type	Employee
Assignment Status	Active

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

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

Immediate Provisioning of Roles

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

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

- Provisioned immediately to all users who do not currently have roles for which they are eligible

- Deprovisioned immediately from users who are no longer eligible for roles that they currently have

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

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

Role-Mapping Names

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

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

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.

Attribute	Value
System Person Type	Employee
Assignment Status	Active

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.

Attribute	Value
Business Unit	Sales
Assignment Status	Active
Manager with Reports	Yes

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.

Attribute	Value
System Person Type	Retiree
Assignment Status	Inactive

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.

Attribute	Value
Department	Sales
Job	Sales manager
Grade	6
Assignment Status	Active

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 Customer Relationship Management (CRM) Self Service, respectively, and associate them with users.

Before Importing Users

Oracle Identity Management (OIM) handles importing users.

If legacy employees, contingent workers, and their assignments exist, the HCM Application Administrator imports these definitions by performing the 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 CRM.

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

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

Import Users

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

If no legacy users or user accounts can be imported or exist in an LDAP repository accessible to Oracle Identity Management (OIM), the IT security manager creates users manually in OIM or uses the 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, CRM or Oracle Fusion Supplier Portal.

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

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

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

Manage Duties

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

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

OIM stores the role hierarchy and the spanning of roles across multiple pillars or logical partitions of applications.

In cases where your enterprise needs to provide access to custom functions, it may be necessary to create or modify the duty roles of the reference implementation.

Tip

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

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

Note

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

Manage Application Access Controls

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

Note

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

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

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. On the Initiate Data 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 import the data to the HCM Data Loader stage tables.

HCM Data Loader is a generic utility for loading data to Oracle Fusion Human Capital Management from external sources.

2. In the Data Exchange work area, you run the Load Batch Data process to load data from the HCM Data Loader 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 has been 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.

Decisions to Consider	In This Example
<p>What are my spreadsheet names?</p> <p>You can define your own naming convention; in this example, the names are selected to make identifying the spreadsheet contents easy.</p>	<ul style="list-style-type: none"> WorkersMMDDYYBatchnn.xlsx WorkersMMDDYYBatchnnErrorsnn.xlsx <p>For example, Workers042713Batch01.xlsx.</p>
<p>What is my batch name?</p>	Workers042713Batchnn
<p>Where will I fix Load Batch Data errors?</p>	In the spreadsheet Workers042713BatchnnErrorsnn.xlsx

Summary of the Tasks

Import worker users by:

1. Selecting the Import Worker Users task
2. Creating the spreadsheet
3. Entering workers in the spreadsheet
4. Importing the spreadsheet data to the HCM Data Loader stage tables
5. Loading workers to the application tables from the HCM Data Loader stage tables
6. Reviewing the results of the Load Batch Data process and correcting errors

Prerequisites

Before you can complete this task, you must have:

1. Installed the desktop client Oracle ADF 11g Desktop Integration
2. Enabled the Trust Center setting **Trust access to the VBA project object** 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.

Field	Name
Search	Task
Name	Import Worker Users

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

The task navigates to the Initiate Data Load page.

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

Creating the Spreadsheet

1. On the Initiate Data 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, because those business objects (regardless of how you create them) must be created before worker users.

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 Workers 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. Enter workers in the spreadsheet.

Ensure that you provide any required values and follow instructions in the spreadsheet for creating additional rows.

Importing the Spreadsheet Data to the HCM Data Loader Stage Tables

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 HCM Data Loader stage tables, its status is updated.
3. When uploading completes, identify any spreadsheet rows with the status Insert Failed, which indicates that the row failed to upload.
4. For any row that failed to upload, double-click the status value to display a description of the error.
5. When you have corrected any errors, click **Upload** again to load the remaining rows to the same batch.

Loading Workers to the Application Tables from the HCM Data Loader Stage Tables

Use the default values except where indicated.

1. In Oracle Fusion Applications, select **Navigator - Workforce Management - Data Exchange**.
2. In the Tasks pane of the Data Exchange work area, click **Load Batch Data**.
3. In the Search region of the Load Batch Data page, enter the batch name Workers042713Batch01 in the **Batch Name** field.

4. Click **Search**.
5. In the Search Results region, select the batch Workers042713Batch01 and click **Run**.
6. On the Schedule Request page, click **Submit**.

Reviewing the Results of the Load Batch Data Process and Correcting Errors

1. On the Load Batch Data page, search again for the batch Workers042713Batch01 and review its status in the **Batch Status** column of the Search Results region.
2. If the batch status is Complete, click **Done** to close the Load Batch Data page; otherwise, continue with the next step.
3. If the batch completed with errors, select the batch in the search results.
4. On the Batch Summary tab of the Batch Details region, select **Action - Extract Failed Objects to Spreadsheet**.

An errors spreadsheet with a standard name is created automatically.

5. Save the errors spreadsheet locally as Workers042713Batch01Errors01.xlsx.

Leave the batch name in the errors spreadsheet as Workers042713Batch01.

6. Review the error messages in the spreadsheet and correct the errors.
7. Repeat this process from the task Importing the Spreadsheet Data to the HCM Data Loader Stage Tables until all errors are fixed.

If further errors occur, increment the errors-spreadsheet suffix by 1; for example, Workers042713Batch01Errors02, Workers042713Batch01Errors03, and so on.

8. Click **Done** to close the Load Batch Data page.

To load a new batch of workers 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.

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

Prerequisites

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

Attribute	Value
System Person Type	Employee
Assignment Status	Active

In the role mapping you include the:

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

Creating a User

1. On the Search Person page, click the **Create** icon to open the Create User page.
2. Complete the fields, as shown in this table:

Field	Value
Last Name	Williams
First Name	Gail
E-Mail	gail.williams@vision.com
Hire Date	4/12/11

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:

Field	Value
Legal Employer	Vision Corporation
Business Unit	Vision Canada

Department	Human Resources
------------	-----------------

Assigning Roles to the User

1. Click **Autoprovision Roles** to provision the employee role to the user.
2. Click **Add Role**.
3. Search for and select the **Expense Claims Administrator** role.
4. Click **Save and Close**. The user account is created and the roles are assigned to the user immediately.

User Details System Extract Report

The Oracle BI Publisher User Details System Extract Report includes details of some or all Oracle Fusion Applications user accounts.

To run this report, you must have an HCM data role that provides view-all access to person records for the Human Capital Management Application Administrator job role.

To run the report:

1. Navigate to **Tools - Reports and Analytics**.
2. In the Contents pane of the Reports and Analytics work area, navigate to **Shared Folders - Human Capital Management - Workforce Management - Human Resources Dashboard**.
3. Select the User Details System Extract report.
4. In the report window, click **More**.
5. On the Oracle Business Intelligence page for the report, select **Open** to run the report immediately or **Schedule** to schedule the report.

Parameters

User Population

Enter one of the following values to identify the group of user accounts to include in the report.

Value	Description
HCM	User accounts with an associated HCM person record.
TCA	User accounts with an associated TCA party account.
OIM	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.
ALL	HCM, TCA, and OIM users accounts.

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.

Value	Description
A	Include active accounts, which belong to users with current roles.
I	Include inactive accounts, which belong to users with no current roles.
All	Include both active and inactive user accounts.

Report Results

The output is an XML-formatted file where user accounts are grouped by type, as follows:

- Group 1 (G_1) includes HCM user accounts.
- Group 2 (G_2) includes TCA party user accounts.
- Group 3 (G_3) includes OIM user accounts.

The information provided in the extract varies with the account type.

HCM User Accounts

Business Unit Name

The business unit from the primary work relationship.

Composite Last Update Date

The date when any one of a number of values, including assignment managers, location, job, and person type, was last updated.

Department

The department from the primary assignment.

Worker Type

The worker type from the user's primary work relationship.

Generation Qualifier

The user's name suffix (for example, Jr., Sr., or III).

Hire Date

The enterprise hire date.

Role Name

A list of roles currently provisioned to workers whose work relationships are all terminated. This value appears for active user accounts only.

Title

The job title from the user's primary assignment.

TCA User Accounts

Organizations

A resource group.

Roles

A list of job, abstract, and data roles provisioned to the user.

Managers

The manager of a resource group.

OIM User Accounts

Start Date

The date from when the account existed.

Created By

The user name of the user who created the account.

FAQs for Manage Users

What happens if I send the user name and password?

An e-mail containing the user name and password is sent to the user's primary work e-mail address. If the user has no primary work-email address, then the

user name and password are sent to the primary work e-mail address of the user's line manager, if available; otherwise, no notification is sent.

You can select **Send user name and password** only if these details have not already been sent for this user: the user name and password can be sent once only for any user. If this option is available for selection but you do not select it, then you can run the process Send User Name and Password E-Mail Notifications later to notify users of their user names and passwords.

Can I extract details of all Oracle Fusion Applications users?

Yes. The Oracle BI Publisher User Details System Extract report includes details of all user accounts or a specified subset. For example, you can produce a report showing inactive user accounts, accounts created between specified dates, or accounts associated with TCA parties only.

To run the report, you must have an HCM data role that provides view-all access to person records for the Human Capital Management Application Administrator job role.

Common Applications Configuration: Define Automated Governance, Risk, and Performance Controls

Segregation of Duties: Explained

Segregation of duties (SOD) separates activities such as approving, recording, processing, and reconciling results so an enterprise can more easily prevent or detect unintentional errors and willful fraud. SOD policies, called access control policies in Application Access Controls Governor (AACG), exert both preventive and detective effects.

SOD policies constrain duties across roles so that unethical, illegal, or damaging activities are less likely. SOD policies express constraints among roles. Duty role definitions respect segregation of duties policies.

Application Access Controls Governor

You manage, remediate, and enforce access controls to ensure effective SOD using the Application Access Controls Governor (AACG) product in the Oracle Enterprise Governance, Risk and Compliance (GRC) suite.

AACG applies the SOD policies of the Oracle Fusion Applications security reference implementation using the AACG Oracle Fusion Adapter.

AACG is integrated with Oracle Identity Management (OIM) in Oracle Fusion Applications to prevent SOD control violations before they occur by ensuring SOD compliant user access provisioning. SOD constraints respect provisioning workflows. For example, when provisioning a Payables role to a user, the SOD policy that ensures no user is entitled to create both an invoice and a payment prevents the conflicting roles from being provisioned. AACG validates the request to provision a user with roles against SOD policies and provides a remediating response such as approval or rejections if a violation is raised.

Use AACG to for the following.

- Define SOD controls at any level of access such as in the definition of an entitlement or role.
- Simulate what-if SOD scenarios to understand the effect of proposed SOD control changes.

- Use the library of built-in SOD controls provided as a security guideline.

Managing Segregation of Duties

SOD policies express incompatible entitlement or incompatible access points into an application. In GRC, an access point is the lowest level access for a particular application. In GRC, entitlement is a grouping of access points. As a security guideline, group the lowest level access points or define the SOD policy at the access level causing the least amount of change. Business activities are enabled at access points. In Oracle Fusion Applications, the hierarchy of access points in descending levels is users, roles, and entitlement.

Note

AACG entitlements are logical groupings of security objects that represent Oracle Fusion Application access points such as roles or entitlement.

Note

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

Oracle Fusion Applications does not predefine business logic for dealing with SOD conflicts. Oracle Fusion Applications does define a set of states where role requests are suspended pending resolution of SOD violations the role request introduces. In most cases, Oracle Fusion Applications invokes OIM to handle role requests. Enterprises define SOD resolution rules when defining SOD policy.

Remediating Segregation of Duties Policy Violations

The risk tolerance of your enterprise determines what duties must be segregated and how to address violations.

AACG assists in remediation of violations with a guided simulation that identifies corrective action. You determine the exact effects of role and entitlement changes prior to putting them into production, and adjust controls as needed.

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

Segregation of Duties in the Security Reference Implementation: Explained

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

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

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

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

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

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

Note

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

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

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

Application Access Controls Governor (AACG)

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

- Define SOD controls at any level of access such as in the definition of an entitlement or role.
- Simulate what-if SOD scenarios to understand the effect of proposed SOD control changes.
- Use the library of built-in SOD controls provided as a security guideline.

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

Conflicts

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

Tip

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

Violations

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

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

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

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

Defining Segregation of Duties Policies: Points To Consider

Segregation of duties (SOD) policies express incompatibilities enforced to control access in defined contexts.

In Oracle Fusion Applications, SOD policies protect against the following incompatibilities.

- Privilege X is incompatible with privilege Y
- Role A is incompatible with role B
- Any privileges in role A are incompatible with any privileges in role B.
- Privilege X is incompatible with any privileges in role B.

The following examples of SOD policies illustrate incompatible entitlement.

- No user should have access to Bank Account Management and Supplier Payments duties.

- No user should have access to Update Supplier Bank Account and Approve Supplier Invoice entitlement.

Data Contexts

You can extend SOD policies to control access to specific data contexts.

For example, no single individual must be able to source a supplier in a business unit and approve a supplier invoice in the same business unit.

Exclusion and Inclusion Conditions

SOD policies may include exclusion conditions to narrow the SOD scope and reduce false positive violations, or inclusion conditions to broaden the scope.

Conditions apply to access points globally, to policies, or to access paths defined by policies. Access path conditions can exclude a user from a role, an Oracle Fusion Applications entitlement from a role, or a permission from an Oracle Fusion Applications entitlement.

The following global exclusion conditions are predefined in Oracle Fusion Applications and available when creating SOD policies.

- User Status
- User Name
- Enterprise Role
- Action
- Business Unit
- Within Same Business Unit

Enforcement

Oracle Fusion Applications enforces SOD policies under the following circumstances.

- When granting entitlement to a role
- When provisioning a role to a user

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

Note

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

A single SOD policy can include entitlement from multiple instances of a single enterprise resource planning environment. For example, one SOD policy is enforced in implementation, test, and production instances of Oracle Fusion Applications.

Managing Segregation of Duties Risks and Violations: Critical Choices

You assess and balance the cost of duty segregation against reduction of risk based on the requirements of your enterprise.

The types of people who resolve SOD conflicts include the following.

- Administrator of an external program such as the Procurement Administrator for the supplier portal or the Partner Manager for the PRM Program
- Senior executive spanning multiple organizations in an enterprise with opposing interests
- Risk management professional implementing an Oracle Enterprise Governance, Risk and Compliance (GRC) initiative
 - Predefines a set of conditions and informs access provisioning staff to approve requests and prove the exception based on certain conditions
 - Allows defining rules to route SOD violations for approval

You view and respond to risks and violations in the Application Access Controls Governor (AACG).

You may wish to override an SOD violation. For example, the Accounts Payable Supervisor includes incompatible duties to create both invoices and payments. When you provision this job role to a user, you may waive the violation in the AACG. You may waive the violation for the currently provisioned user, for the SOD policy that raised the violation, or for the SOD policy within a particular data set, such as a business unit.

The risk tolerance of your enterprise guides how you respond to conflicts. For example, a user may be provisioned with both the role of Order Manager and Shipping Agent. The Order Manger role entitles the user to enter orders, which could result in exploitation when filling shipping quotas. You can remove the entitlement to enter orders that the Order Manger job role inherits from the Orchestration Order Scheduling Duty role. Or you could segregate the shipping and order entry duties by defining an SOD policy that allows a user to have either job role but not both.

False Positives

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

You can reduce false positives by the following methods.

- Define exclusion conditions that can be applied to individual or groups of policies.

- Define logically complex SOD policies that enforce more exacting specifications.
- Determine whether conflicts should be prevented, monitored, or subjected to approval during provisioning.

Path Level Detection

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

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

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

Waiving or Accepting Violations

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

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

Resolving Conflicts

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

The following approaches resolve conflicts.

- Change the segregation of duties policy.
- Ensure a job role does not contain incompatible duties.
- Define data security policies that restrict authorized access by incompatible duties.

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

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

Role Provisioning and Segregation of Duties: How They Work Together

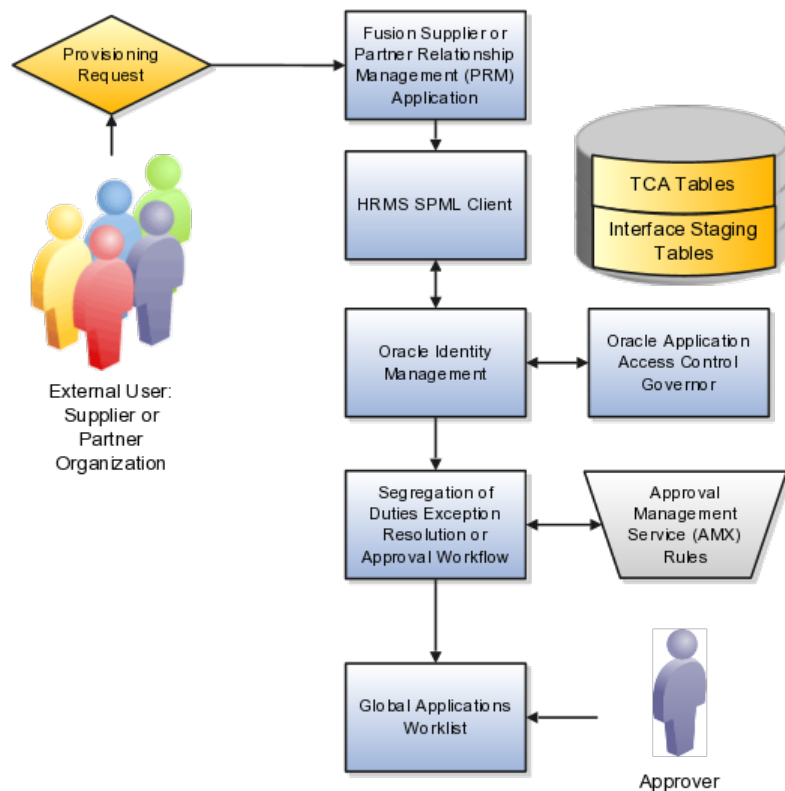
Segregation of duties (SOD) checks occur when roles are assigned to users. The checks are based on Oracle Application Access Controls Governor (AACG) policies in Oracle Enterprise Governance, Risk and Compliance (GRC). The Oracle Identity Management (OIM) integration includes predefined routing rules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

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

Note

In Oracle Identity Management (OIM), external users means users who are not specific to applications, such as enterprise roles or the absence of entitlement to access an application.

The figure shows the role provisioning request flow. OIM uses AACG to check segregation of duties violations.



Tables

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

- Oracle Fusion Trading Community Model
- Interface Staging

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

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

OIM and the SPML Client

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

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

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

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

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

Status	Conflicts	Current State
SOD_CHECK_IN_PROGRESS	Unknown	Request sent to AACG and waiting for response
SOD_REMEDIATION_IN_PROGRESS	Conflict found	AACG detected violations and remediation is in progress
SOD_CHECK_APPROVED	No conflict found	No SOD violations found
SOD_CHECK_REJECTED	Conflict found	AACG detected violations that cannot be remediated
SOD_REMEDIATION_APPROVED	Conflict found	AACG detected violations that are approved

SOD_REMEDIATION_REJECTED	Conflict found	AACG detected violations that are rejected by approver
--------------------------	----------------	--

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

Note

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

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

Oracle Application Access Controls Governor

AACG may respond with the following.

- Roles may be provisioned to the external user or its employees because no SOD conflict is found
- SOD conflict is found and request is denied because the relevant SOD policy is to be strictly enforced and no exception approval should be allowed
- SOD conflict is found and the exception to the policy is allowed, so the request goes through additional processing, such as an approval process.

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

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

Segregation of Duties Exception Resolution or Approval Workflow

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

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

- Organizational hierarchies
- Multiple mandatory and optional approvers

- Rerouting and approval delegation

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

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

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

- Approve the request as it is
- Reject the request

If the approver approves the request, OIM sends an `SOD_REMEDIATION_APPROVED` status to the SPML client.

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

Remediation Task Assignments

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

1. If the role requested is Chief Financial Officer, the SOD remediation task is assigned to the IT Security Manager role.
2. If the SOD violation results from a policy where the SOD control tag is the Information Technology Management business process and the control priority is 1, the SOD remediation task is assigned to Application Administrator role.
3. In all other scenarios, the SOD remediation task is assigned to the Controller role.

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

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

Common Applications Configuration: Define Approval Management for Customer Data Management

Approval Management: Highlights

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

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

Overview

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

See: Introduction to Approval Management

See: Understanding Approval Management Concepts

Approval Groups and Task Configuration

- An approval group consists of a name and a predefined set of users configured to act on a task in a certain pattern. Refer to the Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management.

See: Administering Approval Groups

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

See: Using Task Configuration

- To configure a predefined approval policy, select the predefined rule set and click the Edit task icon button.
- To disable a predefined rule set, select the Ignore this participant check box for that rule set.
- To edit the rules within a predefined rule set, you can insert, update, or delete from the seeded rules as needed while in edit mode.
- You can configure a specific rule to automatically approve without being sent to any approver. Modify the routing for that rule so that it is sent to the initiator (which means the requestor is the approver), set the Auto Action Enabled option to True, and enter APPROVE in the Auto Action field.

Customization

- You can optionally customize predefined approval workflows, for example add post-approval activities or additional stages. Refer to the Oracle Fusion Applications Extensibility Guide.

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. You can also customize help that is embedded in the application, for example hints and help windows, using other tools such as Oracle JDeveloper and Oracle Composer.

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

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 the Oracle Fusion Applications Technology Library. If you select this feature, then these links open the library on the Oracle Technology Network Web site. If you do not select this feature, then your system administrator must download the library from Oracle Technology Network (<http://www.oracle.com/technetwork/documentation/fusion-apps-doc-1508435.html>) and put all the content from within the extracted folder (for example E28271_01) 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.

Only users with the appropriate administrator role can moderate discussions. For more information on granting the administrator role on the discussions server, see the Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal.

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.

Decisions to Consider	In This Example
What type of users do you need to limit help access to?	Human resources (HR) specialists

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

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

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

Field	Value
Help Security Group	HR
Meaning	HR Only
Description	Viewing by HR specialists only
Display Sequence	1

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**.

Help File Customization

Help File Customization: Overview

If you have the appropriate job roles, then you can customize the help files in the help site. Use the Manage Custom Help page to maintain both predefined and custom help files. You can create, duplicate, edit, and delete custom files, or set their status to Active or Inactive. For predefined files, you can only duplicate them or set their status. For each help file, predefined or custom, use help

locations to determine where the help file appears in the application and in the help site. You have various options in how you add custom help, for example by uploading a file or specifying a URL.

Note

To make a copy of all custom help for testing, migration, or other purposes, use the export and import feature in the Setup and Maintenance work area.

There are various ways to access help customization.

- Many help files can be accessed from help windows in the application. If you want to customize help in the context of a help window, for example create a custom help file and add a link to it from a specific help window, then start by opening that help window. When you click the **Manage Custom Help** link, you go to the Manage Custom Help page, and the help location fields are automatically populated with values that correspond to the help window. This way you can easily select existing files to add to the same help location, and when you create a new file, the same help location appears by default.
- Open the Manage Custom Help page directly from the home page of Oracle Fusion Applications Help or from search result pages.
- To edit a specific file, you can either find it in the Manage Custom Help page, or open the file itself and click the **Edit** link.
- Likewise, you can find glossary terms in the Manage Custom Help page, or click the **Glossary** link in the global area to open the Glossary tab, search for the term, and click **Edit**.

Note

When you search in the Manage Custom Help page, make sure that the **Custom Help Only** check box is not selected if you are looking for predefined help.

If your enterprise has purchased Oracle User Productivity Kit (UPK) content, then your administrator can also add a **UPK** item to the **Help** menu in the global area of Oracle Fusion Applications. When users select this menu item, they access UPK content specific to the page that they are on.

Help Types: Explained

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

Demo

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

Example

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

FAQ

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

Glossary

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

Help Topic

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

PDF Guide

PDF guides present a collection of help content from the other help types, except demos, in an organized and logical format. For example, there are guides addressing specific business processes and setup offerings. You can see lists of all guides from the **Guides** menu in Oracle Fusion Applications Help.

Help Locations: Explained

Help locations determine where users can find help files, custom or not, from either the application or the help site.

Help locations include:

- Page or section values
- Help hierarchies
- Primary locations

Page or Section Values

The value in the **Page or Section** field on the help customization pages represents where users can click a help icon to open a help window that contains

a link to the help file. In most cases, this value represents a page or region header in the application. Help windows are also available on specific tabs or windows, and in the Setup and Maintenance work area for specific task lists or tasks. You can associate a help file with multiple page or section values, or with none at all.

The page or section value reflects the logical navigation to the help window. For example, **Edit Opportunity page, Revenue tab, Recommendations window** does not mean that the help file is available in three different places. The help icon is in the Recommendations window, which is accessed from the Revenue tab on the Edit Opportunity page.

If the value suggests multiple locations, for example **Create and Edit Opportunity pages**, then the help file is available from the page header of both the Create Opportunity and Edit Opportunity pages. If the page or section value is, for example, a dashboard region that appears in multiple dashboards, then the value does not specify the page name but just the region. The help file is available from that region in multiple dashboards.

Help Hierarchies

Help files are associated with help hierarchies, which are used to categorize help files and aid users in finding help. Each help file can have multiple hierarchies, with at least one of type Business Processes. The business process hierarchy is based on the Business Process Management model. Every page or section value is predefined with a specific business process hierarchy. If you select a page or section without entering a business process hierarchy, the predefined hierarchy appears by default.

The Search by Business Process navigator in the help site is based on the business process hierarchy. For example, if you assign two business process hierarchies to a help file, users can find the file in both locations in the navigator. When the user clicks **More Help** from a help window, all help files assigned to the same business process hierarchy as the page or section value are returned as search results.

Similarly, the Search by Product navigator is based on the Product hierarchy type, in which level 1 is the product family, level 2 is the product, and level 3 is the business activity owned by that product.

The Search by Functional Setup navigator is based on the Functional Setup hierarchy type. The level 1 nodes for this hierarchy are:

- Functional Setup Manager, which includes help about using the Setup and Maintenance work area.
- Guides, which contains level 2 nodes that correspond to business areas and setup offerings. All the user reference and functional setup PDF guides are included.
- Offerings, which contains level 2 nodes for each setup offering, and lower levels for the main task lists in the offerings. Help for the task lists and tasks are included.

The Search by Common Tasks navigator is based on the Welcome hierarchy type. The level 1 nodes represent categories of functional areas common to all users.

Primary Locations

The primary location of a help file designates the hierarchy that is displayed for the help file in search results and within the help content as breadcrumbs. You cannot change the primary location of a help file that came with your help installation. Primary locations of predefined help are based on the business process hierarchy, while custom help files can have primary locations based on hierarchies of any type.

Editing Predefined Help and Glossary Terms: Points to Consider

When you open any predefined help file, including glossary terms, that came with Oracle Fusion Applications Help, you can see an edit option if you have roles allowing edit access. When you edit predefined help, keep in mind:

- What happens to the original help file
- Where predefined help appears
- Considerations specific to glossary terms

What Happens to the Original Files

When you edit predefined help, you are actually creating a new custom help file based on the original file, with the same help locations. The customized version replaces the original, which becomes inactive and hidden from users. You can display both versions by reactivating the original in the Manage Custom Help page.

Note

In the Search Results: Existing Help region on the Manage Custom Help page, there is no option to edit predefined help. You can duplicate a predefined help file, edit the copy, and optionally inactivate the original.

Where Predefined Help Appears

All predefined help comes with preassigned help locations, including at least one based on the hierarchy of type Business Processes. Many also have predefined page or section values that indicate where the help can be accessed from help windows in the application.

To change where predefined help appears, either in the help site navigators or in the application, create a duplicate in the Manage Custom Help page. Change or add help locations to your custom copy, and inactivate the original.

Even though glossary terms do not appear in the help site navigators, you still need to enter at least one help location to categorize the glossary term.

Considerations Specific to Glossary Terms

When you edit a predefined glossary term, the original term becomes inactive. Existing links to the glossary term, from other predefined and custom help files, will automatically point to your custom version. If you later inactivate the custom glossary term, make sure to activate the original term so that the links still work.

Links in Custom Help: Points to Consider

When you create or edit custom help, follow best practices when you include links to help files or other content. If you are working on custom help created by duplicating a predefined help file, then you may see existing links from the original file in the Help Content section. The types of links that you can work with include:

- Related help links
- Standard hypertext links
- Links to documentation library content
- Glossary term links

For all link types, except the standard hypertext links, you must create or edit custom help with a Text or Desktop source type. In other words, you must type the help content directly in the application or use an HTML file that you upload to help. For standard hypertext links, the source type can also be URL.

Related Help Links

Related help is the section at the end of help files that contains links to other help files. The syntax for related help contains a comma-separated list of title IDs that represent help files.

This figure provides an example of related links code.

```
OfaRelatedTopics(CREATE_AUTOMATIC_POSTING_CRITERIA_S_0000,  
JOURNAL_ENTRIES_HOW_THEY_RE_RECORDE_0000)
```

- You can delete this code to remove all related help, or delete title IDs to remove individual links (for example, CREATE_AUTOMATIC_POSTING_CRITERIA_S_0000).
- To replace existing links or add new links, you need to retain the code syntax and enter desired title IDs. To find title IDs, search for the help files on the Manage Custom Help page. Title IDs are displayed in the search results, but the **Title ID** column is hidden by default.

Standard Hypertext Links

You can create standard hypertext links to any file or Web site as long as you ensure the stability and validity of the links, including links to other help files,

custom or not. These links can appear anywhere in the body of your help file as long as they come before any related help links.

In the Help Content section, highlight the text that you want to use as link text and click the **Add Link** icon button.

For links to other help files, open the file to which you want to link, and click the **E-Mail** link. Use the URL in the autogenerated e-mail text as the link to the file.

Note

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

Links to Documentation Library Content

The syntax for links to HTML files in documentation libraries, for example the Oracle Fusion Applications Technology Library, is:

```
<span class="HP_topic-link_bridgeDocument-linkToSTDoc_"><?ofa  
linkToSTDoc(WCSUG4636) ?><span class="HP_topic-linktext_">Understanding  
Tags</span><?ofa endLink ?></span>.
```

WCSUG4636 is the anchor ID and Understanding Tags is the link text. You can:

- Modify the link by replacing the existing anchor ID or editing the link text, or both.
- Remove the link by deleting all the code for it.
- Create links to documentation library content by following the same syntax. These links can appear anywhere in the body of your help file as long as they come before any related help links.

Important

To ensure that you are linking to a supported documentation library, enter anchor IDs only from documentation libraries that are linked from predefined help topics.

Glossary Term Links

Glossary term links provide definitions in a note box when users hover over the term in help files.

This figure shows an example of code for a glossary term link.

```
OfaGlossaryTerm("accounting period", ACCOUNTING_PERIOD_0001)
```

In this example, `accounting period` is the link text, or glossary term, and `ACCOUNTING_PERIOD_001` is the identifier, or title ID.

- To remove the link but retain the text, delete all the code except the term itself.

- To add glossary term links, you must follow the link syntax and use the correct title ID for the glossary term. You can find title IDs in the search results of the Manage Custom Help page.

Note

If your custom help has glossary terms and the source type is Desktop File, then make sure before uploading that the quotes around the glossary term are actual quotation marks in raw HTML, not ". Otherwise, quotation marks will appear when users view the help file.

Customizing PDF Guides: Worked Example

This example demonstrates how to customize a PDF guide that came with Oracle Fusion Applications Help. This guide is currently not available from any help window in the application.

The following table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
What changes do you need to make to the guide?	Change the title of a chapter and remove a section in that chapter, to hide content about a particular subject
Which help window should the customized guide appear in?	The help window for the entire Welcome dashboard of Oracle Fusion Applications
Which help navigators should the customized guide appear in, and on which node?	Same as the original guide, plus the path associated with the help window
Do you want to limit access to the customized guide?	No, same as the original guide

Edit a copy of the original PDF guide, and use the Manage Custom Help page to replace the original PDF guide with your new file.

Copying and Editing the PDF Guide

1. Open the original PDF guide from the help site and save a copy to your desktop. Leave open the help file for the guide.
2. Using a PDF editor application, change the title of the chapter wherever the chapter title appears. Delete the content you want to hide from users.
3. Make sure that your new PDF guide is less than 6 MB.

Replacing the Original PDF Guide

1. In the help file that you still have open for the original PDF guide, click the **Edit** link.
2. On the Create Help page, use the default values except where indicated.
3. Update the title to the name that you want to display to users.
4. In the **File Name** field, browse for and select your customized guide.

5. Delete any keywords or parts of the description relevant to the content you removed from the PDF guide.
6. Add a help location with the Business Processes hierarchy type and select **Information Technology Management** as the level 1 node, **Manage Enterprise Application Software** as the level 2 node, and **Use Applications** as the level 3 node.
7. Select **Welcome page** in the **Page or Section** column.
8. Click **Save and Close**. The help file for the original PDF guide is automatically set to inactive.

Adding Custom UPK Content to Help: Worked Example

This example demonstrates how to add custom Oracle User Productivity Kit (UPK) topics as demo help files. These help files function like any predefined help file for demos. You can search and include these files in help windows and navigators as you would other help.

In this scenario, you are adding two demos about social networking, to appear in help windows on the Welcome dashboard.

Note

Your demo must be made with UPK 3.6.1 or later to be added as help.

The following table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
What UPK content do you want to add to help?	From a UPK module containing five topics, add two as custom demos on the help site
Which help navigators should each demo appear in, and on which node?	Because the two demos are about social networking: <ul style="list-style-type: none"> • Search by Common Tasks navigator, under the Collaboration node • Search by Business Process navigator, under Information Technology Management - Manage Networking and Communications - Manage Social Networking Capabilities
Which help window should each demo appear in?	On the Welcome dashboard of Oracle Fusion Applications, one demo goes in the help window in the Activity Stream region, and the other in the People Connection region
Do you want to limit access to the help files for the demos?	No
Do you want the help files to appear in the New and Updated pane?	Yes

Generate a report of UPK document IDs, which you will use when creating custom help, to identify the UPK topics that you want to add. Publish the UPK module as a player package, then create custom help for the UPK topics that you want to use as help demos.

Generating a UPK Document ID Report

1. In the UPK Developer, select **Details View**.
2. Right-click any column header, for example Name, and select **Column Chooser**.
3. In the Column Chooser dialog box, click and drag the Document ID column header and drop it after the Name column. Close the Column Chooser dialog box.
4. From the File menu, select to print, and save the output as a Microsoft Excel file to your desktop.

Creating the Player Package

1. From the UPK Developer, make sure that the topics that you want to add as demos have the See It play mode. The topics can also have other modes, but only the See It mode is included in the custom help file.
2. Publish the module, specifying any location for the output and selecting to publish the selection only.
3. In the Formats section of the Publish Content window, select the **Player** check box under the **Deployment** check box group.
4. In the Player section, select the **Include HTML Web Site** check box, to ensure that the custom help file includes a text-only version of the UPK topic.
5. Finish the publishing process, after optionally setting other options.
6. Navigate to the location where you specified the output to be generated.
7. In the Publishing Content folder, copy the PlayerPackage folder and add it to the web server where you store UPK content.

Creating Custom Help for Demos

1. Open the help window in the Activity Stream region on the Welcome dashboard of Oracle Fusion Applications, and click **Manage Custom Help**.
2. On the Manage Custom Help page, the page or section and hierarchy values are populated with the values for the Activity Stream region.
3. Click **Create**.
4. On the Create Help page, complete the fields in the General Information section, as shown in this table. Use the default values except where indicated.

Field	Value
Title	The name of the UPK topic.
Source Type	Oracle User Productivity Kit
File Location	The full URL of the player package folder on the Web server, for example, <code>http://<your domain>.com/UPKcontent/PlayerPackage.</code>

Document ID	The document ID of the UPK topic to add to the help window in the Activity Stream region. You can copy and paste this ID from the Microsoft Excel file that you generated earlier.
Help Type	Demo
Help Security Group	Unsecured
Keywords	Terms relevant to the demo.
Description	Summary of the demo.
Include in New and Updated pane	Selected

The Help Location section contains values for the help window in the Activity Stream region. This help file will also appear in the Search by Business Process navigator under this predefined hierarchy.

5. Click **Save and Close**.
6. On the Manage Custom Help page, open the help locations for the help file that you just created.
7. Add a help location with the Welcome hierarchy type and select **Collaboration Features** as the level 1 node.
8. Add another help location with the Business Processes hierarchy type and select **Information Technology Management** as the level 1 node, **Manage Networking and Communications** as the level 2 node, and **Manage Social Networking Capabilities** as the level 3 node.
9. Click **Save and Close**.
10. Starting at the Connections region, repeat steps 1 to 9 for the other UPK topic that you want to add.

FAQs for Help File Customization

How can I restrict help content to specific user roles?

When you create or edit help, select a help security group that represents the set of roles that you want to have access to the help. If you do not see the Security Group field, then your administrator has not selected the Custom Help Security feature choice. The Unsecured group has no associated roles, so anyone can view the help. The predefined Secured group includes all internal employees and contingent workers, unless this group has been edited. You can create security groups and associate roles using the Manage Help Security Groups page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Help Security Groups task. Your new security groups are immediately available for use to secure new or edited help files.

Why can't I select and add help to a location?

You must specify a page or section to add the existing help to. To ensure that help is added to the correct help window, go to the page or section in the

application, click the **Help** icon, and click the **Manage Custom Help** link in the help window. Alternatively, in the Manage Custom Help page, search for at least a page or section and a level 1 value for the Business Processes hierarchy type before selecting the **Select and Add** option.

You cannot select and add help to a particular hierarchy, on the Manage Custom Help page, without a page or section. To add just a hierarchy, search for the help file, add a new help location, and specify only the hierarchy information.

What happens to custom help when a help patch is applied?

Oracle Fusion Applications Help patches update all help files, both active and inactive, except custom help. Custom help files are not affected by patches. Consider reviewing inactive files to see if you want to activate the updated version, or to make similar edits to the custom versions of those files, if any.

Embedded Help Customization

Customizing Embedded Help: Highlights

You can customize help that is embedded in the application, for example hints and help windows, for all users of Oracle Fusion Applications.

Embedded help customization is fully described in the Oracle Fusion Applications Extensibility Guide.

- Edit, create, or delete hint text that appears on hover over buttons, links, icons, or tab titles.

See: Customizing or Adding Bubble Embedded Help

- Edit, create, or delete other types of embedded help. Refer to the Customizing or Adding Static Instructions, In-Field Notes, and Terminology Definitions section.

See: Oracle Fusion Applications Extensibility Guide

Common Applications Configuration: Define Application Toolkit Configuration

Define Application Toolkit Configuration: Overview

Oracle Fusion Application Toolkit (ATK) is an application that provides various core components of Oracle Fusion Applications, including the Welcome dashboard, Oracle Fusion Applications Help, the Reports and Analytics pane, and the Watchlist feature. Use the Define Application Toolkit Configuration task list to set up and maintain some of these components for all users, and the Define Help Configuration task list for Oracle Fusion Applications Help.

Note

The Define Application Toolkit Configuration task list is available in implementation projects only if the Application Toolkit Component Maintenance feature choice is selected.

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

Map Reports to Work Areas

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

Aside from determining which work areas a specific report is mapped to, you can perform additional setup for reports in the context of the Reports and Analytics pane. You can set up report permissions, and enable Oracle Business Intelligence (BI) Publisher reports for scheduled submission.

This additional setup is described in the Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition) and the Oracle Fusion Applications Extensibility Guide.

Report Permissions

- You can restrict access to specific reports for specific users, and this security is not limited to the Reports and Analytics pane. Refer to the

Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

See: Assigning Permissions

Oracle Business Intelligence Publisher Reports Submission

- Oracle BI Publisher reports must be registered as processes with Oracle Enterprise Scheduler to be enabled for scheduling. This registration also enables a Schedule link for the report in the Reports and Analytics Pane. Refer to the Oracle Fusion Applications Extensibility Guide, and perform the following steps in the specified order.
 - Create an Oracle Enterprise Scheduler job definition for the report.
See: Tasks Required to Run Custom Reports with Oracle Enterprise Scheduler Service
 - Specify the job definition details in the report's properties.
See: Enabling Reports for Scheduling from the Reports and Analytics Pane

FAQs for Map Reports to Work Areas

How can I set up the Reports and Analytics pane for all users?

You can remove any currently mapped report from the Reports and Analytics pane, or add mappings to reports from the Oracle Business Intelligence (BI) Presentation catalog. To access the setup, click **Edit Settings** in the Reports and Analytics pane, or use the Map Reports to Work Areas task in the Setup and Maintenance work area. If you do the former, then you can set up only the Reports and Analytics pane on the work area that you are in.

If you do the latter, then you can select a work area to set up. If you do not see the desired work area, most likely you do not have access to it due to security. You can request to be granted a role that has access to the work area, or another administrator or business user with access to the work area can be granted the Reports and Analytics Region Administration Duty to be able to map reports to the work area.

Tip

On the Map Reports to Work Areas page only, you can also use the Synchronize button to remove mappings to reports that are no longer in the catalog, for all work areas at once.

Any changes you make in either UI apply to all users with access to the mapped work area.

Why can't I see reports when mapping reports to work areas for the Reports and Analytics pane?

It is possible that there are no reports currently mapped to the work area that you select in the Map Reports to Work Areas page. Alternatively, reports are mapped, but you do not see them due to security.

Similarly, in the list of all available reports from the catalog, you can see only the reports that you have access to. You can request to be granted a role that has access to the reports that you want to map, or another administrator or business user with access to those reports can be granted the Reports and Analytics Region Administration Duty to be able to map reports to work areas.

Why can't I see reports when I edit settings for the Reports and Analytics pane?

In the Edit Settings window, you may not be able to see a currently mapped report because you do not have access to it due to security.

Similarly, in the list of all available reports from the catalog, you can see only the reports that you have access to. You can request to be granted a role that has access to the reports that you want to map, or another administrator or business user with access to those reports can be granted the Reports and Analytics Region Administration Duty to be able to map reports to work areas.

Set Watchlist Options

Watchlist Setup: Points to Consider

For all users across the site, you can disable or enable predefined Watchlist categories and items, edit their names, and determine how often item counts refresh. You cannot delete predefined Watchlist categories and items, nor create any for the site. Users can create their own Watchlist items through saved searches.

Access the Set Watchlist Options page by starting in the Setup and Maintenance Overview page and searching for the Set Watchlist Options task.

Disabling Predefined Categories and Items

Use the Set Watchlist Options page to enable or disable predefined Watchlist categories and items. Disabling any category or item also disables associated processes involved in calculating the Watchlist item counts for all users. These processes include creating data caches, performing security checks, invoking services across domains, running queries, and so on.

An item with the **Predefined** type represents the actual predefined Watchlist item that appears in the Watchlist. If you disable this type of Watchlist item, then:

- The item is not available for users to display in their watchlist
- The item is removed from any watchlist where it is currently displayed

A Watchlist item with the **User-created saved search** type does not appear in the Watchlist; it controls the display of the **Manage Watchlist** button or menu item in pages with saved searches. If you disable this type of Watchlist item, then:

- The **Manage Watchlist** option is not available to users in the corresponding work area, so users cannot use their own saved searches as

Watchlist items. A message is displayed to users when they try to use this option.

- Any user-defined saved searches from that work area already used as Watchlist items are no longer available in the users' watchlist. The user-defined saved searches are still available to be used for searching, but not for the Watchlist.

If you disable a Watchlist category, then the category is not available for users to include in their watchlist, and all Watchlist items within the category are also disabled.

Ultimately, the Watchlist for any user contains the subset of categories and items that are enabled in the Set Watchlist Options page:

- Plus any items based on user-defined saved searches
- Minus any categories or items that the user chooses to hide using Watchlist preferences
- Minus any items with no results found, if the user chooses to hide such items using Watchlist preferences

Specifying Refresh Intervals

All Watchlist items have a predefined refresh interval, which controls how often the query that calculates the count for a Watchlist item can be run. Use the Set Watchlist Options page to edit the interval values. What you specify as the refresh interval for a Watchlist item of type User-created Saved Search applies to all Watchlist items based on saved searches created by users on the corresponding search page.

When the user is in the Welcome dashboard with the Watchlist open for at least two and a half minutes, the query automatically runs for all Watchlist items if no refresh already ran in this user session. To subsequently run the query again, users can manually refresh the Watchlist region. The **Refresh** icon is enabled after five minutes since the last refresh.

Note

During a refresh, the query runs for an individual Watchlist item only if the time since the last query for this item is equal to or greater than the specified refresh interval. Since the manual refresh of the entire Watchlist is not available until five minutes after the last refresh, you should not set a Watchlist item refresh interval that is less than five minutes.

When users open Watchlist from the global area, a refresh automatically runs if five minutes have passed since the last refresh. During this refresh, the query runs for an individual Watchlist item only if the time since the last query for this item is equal to or greater than the specified refresh interval.

For example, you set the interval to eight minutes for a particular Watchlist item. When the user signs in and goes to the Welcome dashboard, with the Watchlist open, the query automatically runs for this Watchlist item after two and a half

minutes. Every two and a half minutes after, a check is performed for stale counts and new cached counts are displayed.

Five minutes after the query ran, the **Refresh** icon is enabled and the user performs a manual refresh. However, the query does not run for this Watchlist item, because the refresh interval is eight minutes. The user navigates away from the Welcome dashboard and opens the Watchlist from the global area six minutes later. A refresh automatically runs because more than five minutes have passed since the last refresh. This time, the query runs for this Watchlist item because it has been more than eight minutes since the query last ran for this item.

Editing Predefined Category and Item Names

Predefined Watchlist category and item names are stored as meanings of standard lookups. Lookup types for predefined categories end with WATCHLIST, for example EXM_EXPENSES_WATCHLIST. Edit the lookup type meaning to change the category name. To change item names, edit lookup code meanings for that lookup type.

Common Applications Configuration: Maintain Common Reference Objects

Maintain Common Reference Objects: Overview

The Maintain Common Reference Objects task list contains Oracle Fusion 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.

Note

Offerings also include application-specific tasks for managing Applications Core objects. For example, the Financials offering includes tasks such as Manage Receivables Descriptive Flexfields, and Manage Receivables Lookups.

For more information on configuring custom objects, see the Oracle Fusion Applications Extensibility Guide.

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 Fusion Middleware Extensions for Applications.

Names

Module name is the logical name for the module and is always available. The name must be unique among nodes in the same hierarchy level with the same parent, but try to make it as unique in the whole hierarchy as possible.

The user name and description can appear to users in other parts of Oracle Fusion Applications, so make sure that the values are something that users know to represent the module.

Usage Types

Though you can update the usage type to reflect the current state of the module, just doing so does not affect the actual state. For example, setting a module as installed does not mean it is actually installed if the installation itself has

not taken place. Installation refers to operations related to laying down all the components needed to create an Oracle Fusion Applications environment, while deployment is the process that starts the managed servers and clusters and facilitates the actual use of product offerings. A licensed module is available for installation and deployment, and a deployed module is considered actively used when actually used by users.

Seed Data

If seed data is allowed, then seed data such as flexfields and lookups can be extracted for the module using seed data loaders. By default, extract is allowed for all predefined modules of type application and logical business area.

Associations

You can associate a logical domain to modules of type product family, as well as one or more enterprise applications to modules of type application. This association represents the relationship between the taxonomy modules and the corresponding domain and enterprise applications stored in the Oracle Fusion Applications Functional Core (ASK) tables.

Define Reference Data Sharing

Reference Data Sharing: Explained

Reference data sharing facilitates sharing of configuration data such as jobs and payment terms, across organizational divisions or business units. You define reference data sets and determine how the data is shared or partitioned. Use reference data sets to reduce duplication and maintenance by sharing common data across business entities where appropriate. Depending on the requirement (specific or common), each business unit can maintain its data at a central location, using a set of values either specific to it or shared by other business units.

You can share reference data after it is filtered on the basis of sets. A common reference data set is available as the default set, which can be assigned to several business units sharing the same reference data. For commonly used data such as currencies, you can use the common reference data set and assign it to multiple business units in various countries that use the same currency. In cases where the default set cannot be assigned to an entity, you can create specific sets. The data set visible on the transactional page depends on the sharing method used to share reference data.

For example, XYZ Corporation uses the same grades throughout the entire organization. Instead of managers in different business units setting up the same grades, XYZ Corporation decides to create a set called Grades and assign the grades reference data group for all business units in the organization to the Grades set, so that the grades can be shared.

Note

For specific information on configuring reference data sharing for a particular object or product, refer to its product documentation.

Reference Data Sets: Explained

Reference data sets are logical groups of reference data that can be accessed by various transactional entities depending on the business context. Oracle Fusion Applications contains a common reference data set as well as an enterprise set that may be used as a default set. Depending on your business requirement you can create and maintain additional reference data sets, while continuing to use the common reference data set.

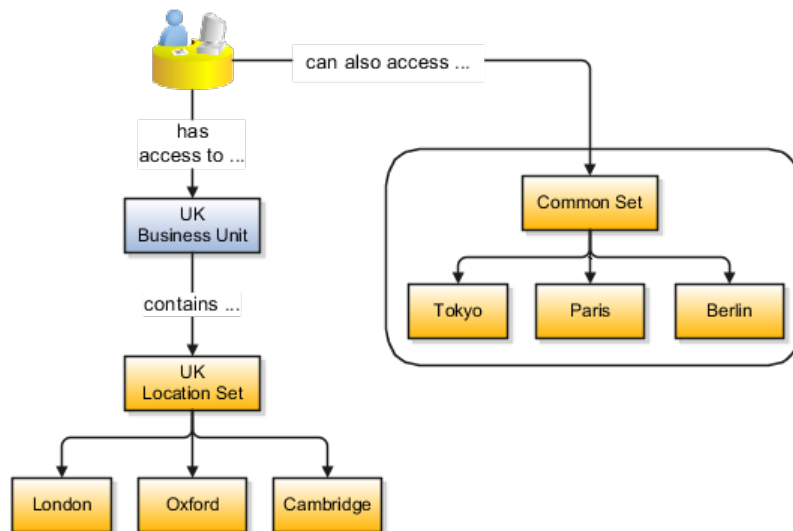
Consider the following scenario.

Your enterprise can decide that some aspects of corporate policy should affect all business units and leave other aspects to the discretion of the business unit manager. This allows your enterprise to balance autonomy and control for each business unit. For example, if your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level, you can let managers define their own sales methods, but define payment terms centrally. In this case, each business unit would have its own reference data set for sales methods, and there would be one central reference data set for payment terms assigned to all business units.

Partitioning

The partitioning of reference data and creation of data sets enable you to create reference entities across tables or lookup types, and share modular information and data processing options among business units. With the help of partitioning, you can choose to create separate sets and subsets for each business unit depending upon its business requirement, or create common sets or subsets to enable sharing reference data between several business units, without the need for duplicating the reference data. Partitioning provides you the flexibility to handle the reference data in a way appropriate to your business needs.

The following figure illustrates the reference data sharing method (assignment to one set only, with common values) where the user can access the data assigned to a specific set in a particular business unit, as well as access the data assigned to the common set.



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.

Type	Description
Asset Book	Information about the acquisition, depreciation, and retirement of an asset that belongs to a ledger or a business unit.
Business Unit	The departments or organizations within an enterprise.
Cost Organization	The organization used for cost accounting and reporting on various inventory and cost centers within an enterprise.
Project Unit	A logical organization within an enterprise that is responsible for enforcing consistent project management practices.
Reference Data Set	References to other shared reference data sets.

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 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.

Lookup Type	Lookup Code	Meaning	Tag
MAR_STATUS	M	Married	
	S	Single	
	R	Registered Partner	+NL
	DP	Domestic Partner	-FR, AU

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.

Allowed Task	User	Extensible	System
Deleting a lookup type	Yes	No	No
Inserting new codes	Yes	Yes	No
Updating start date, end date, and enabled fields	Yes	Yes, only if the code is not predefined data	No
Deleting codes	Yes	Yes, only if the code is not predefined data	No
Updating tags	Yes	No	No
Updating module	Yes	No	No

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.

Lookup	Description
Standard	Lists the available codes and translated meanings
Set enabled	Additionally associates a reference data set with the lookup codes

Common	Legacy lookups
--------	----------------

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.

Lookup type parameters	Value
Lookup type name	COLORS
Meaning	Status
Description	Status by color
Module	Oracle Fusion Middleware Extensions for Applications

The lookup codes you define for the COLORS lookup type are, BLUE, RED, GREEN, and YELLOW.

Lookup Code	Meaning	Enabled	Display Sequence
BLUE	Urgent	No	4
RED	Stop	Yes	1
GREEN	Go	Yes	3
YELLOW	Caution	Yes	2

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.

Displayed Value	Hidden ID
Stop	RED
Caution	YELLOW
Go	GREEN

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.

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

Transaction number	User name	Status
1	Jane	RED
2	Bob	YELLOW
3	Alice	BLUE

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.

Reference Data Set	Lookup Code	Lookup Meaning
US	RED	Red
US	YELLOW	Yellow
US	GREEN	Green
EU	RED	Rouge
EU	ORANGE	Orange

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.

Reference Data Set	Lookup Code	Lookup Meaning
US	SEC	SEC Compliance Review
US	DIR	Needs Director's Approval
US	VP	Needs Vice President's Approval
CHINA	CSRC	Pending China Securities Regulatory Commission Review
CHINA	PR	Needs President's Approval
COMMON	REQUESTED	Customer Request

Using the Manage Set Assignments task, you have defined assignments that designate the China business unit to refer to the CHINA and the US business unit to refer to the US and all business units to refer to the COMMON set. When end users place a contract hold in the US business unit, only the three reason codes in US_SET are available. When placing a contract hold in the China business, only the two codes in China_SET are available.

FAQs for Define Lookups

How can I edit lookups?

You can edit the existing lookup codes of a lookup type or add new lookup codes on the Define Lookups pages, which you can access by starting in the Setup and Maintenance work area and searching for lookup tasks. You can edit the existing lookup codes of a lookup type, or add new lookup codes to a lookup type, if the customization level for the lookup type supports editing.

Why can't I see my lookup types?

Lookups are listed by lookup type. Typically lookup types are managed using tasks that handle a group of related lookups, such as Manage Geography Lookups. Each task gives you access only to certain lookup types. The generic tasks provide access to all lookup types of a kind, such as all common lookups using the Manage Common Lookups task.

If existing lookups are not available to the tasks of the Define Lookups activity, they may be validated for use in a lookup view that is not central to all applications or whose owning application has not been specified in a lookup view.

Lookups can only be managed in the Define Lookups tasks if the lookup's view application is the standard lookups view, common lookups view, or set-enabled lookups view. Lookups defined in an application view can only be managed by following instructions provided by the owning application.

Note

A lookup type and its codes can only be defined in one lookup view.

What's the difference between a lookup type and a value set?

A lookup type consists of lookup codes that are the values in a static list of values. Lookup code validation is a one to one match.

A table-validated value set can consist of values that are validated through a SQL statement, which allows the list of values to be dynamic.

Tip

A table validated value set can be defined based on any table, including the lookups table. This allows a lookup type to be made into a table-validated value set that can be used in flexfields.

Area of Difference	Lookup Type	Value Set
List of values	Static	Dynamic if Table validation type
Validation of values	One to one match of meaning to code included in a lookup view, or through the determinant of a reference data set	By format or inclusion in a table
Format type of values	char	varchar2, number, and so on
Length of value	Text string up to 30 characters	Any type of variable length from 1 to 4000
Duplication of values	Never. Values are unique.	Duplicate values allowed
Management	Managed by both administrators and end-users, except system lookups or predefined lookups at the system customization level, which cannot be modified.	Maintained by administrators, except some product flexfield codes, such as GL for Oracle Fusion General Ledger, which are maintained by end users

A lookup type cannot make use of a value from a value set.

Value sets can make use of standard, common, or set-enabled lookups.

Both lookup types and value sets are used to create lists of values from which users select values.

What's a lookup tag used for?

Tags on lookup codes allow you to add a label to your lookup codes.

Lookup tags are unvalidated and uninterpreted by lookups. A tag can be used to categorize lookups based on facilitating searches or guiding how a lookup should be used.

Document what the tag on a lookup represents and how to use it.

Manage Messages

Messages: Highlights

The message dictionary contains messages that tell users about business rule errors, such as missing or incorrect data, and how to resolve them, to warn users about the consequences of intended actions, and provide information in log files. These messages are defined for specific applications and modules, but a few are common messages that can be used in any application. All applications also use messages stored outside of the message dictionary.

The message dictionary is described in the Oracle Fusion Applications Developer's Guide, and other messages in the Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework.

Managing Messages

- Use the Manage Messages page to create and edit custom messages in the message dictionary, as well as edit predefined messages. Do not delete predefined messages unless you are sure that they are not used anywhere. Refer to the Oracle Fusion Applications Developer's Guide.

See: Introduction to Message Dictionary Messages

- Messages outside of the message dictionary, for example confirmations and field validations, are managed in Oracle Application Development Framework Faces components or through message resource bundles used for translation. Refer to the Oracle Fusion Middleware Web User Interface Developer's Guide for Oracle Application Development Framework.

See: Displaying Hints and Error Messages for Validation and Conversion

See: Internationalizing and Localizing Pages

Creating and Editing Messages: Highlights

Each message in the message dictionary has many attributes and components, including message properties, text, and tokens, that you define when creating or editing the message.

Details about these messages are described in the Oracle Fusion Applications Developer's Guide.

Message Properties

- The message type identifies the type of information that the message contains.

See: Understanding Message Types

- The message name and number are identifiers for the message. There are specific message number ranges for predefined messages in each application, and you should not edit numbers assigned to predefined messages. When creating custom messages, use only message numbers within the 10,000,000 to 10,999,999 range.

See: About Message Names

See: About Message Numbers

- The translation notes for predefined messages might contain internal content that you can disregard.

See: About Translation Notes

- The message category, severity, and logging enabled option are related to the incident and logging process.

See: About Grouping Messages by Category and Severity

See: Understanding Incidents and Diagnostic Logs with Message Dictionary

Message Text and Tokens

- The message text comprises various components, some of which are displayed only to select users. To determine which component of the message text is displayed to a particular user, set the Message Mode profile option (FND_MESSAGE_MODE) at the user level for that user. The message component short text is visible to all users and therefore, the profile option does not apply to this component. Also, the profile option applies only to messages in the message dictionary.

See: About Message Components

- Tokens are variables that represent values to be displayed in the message text.

See: About Tokens

Common Messages: Points to Consider

Common messages, which have message names that begin with FND_CMN and message numbers between 0 and 999, are used throughout Oracle Fusion Applications. Each common message can appear in multiple places in any product family. For example, the FND_CMN_NEW_SRCH message can be used for any search to indicate that no results were found. Common messages that are of type error or warning are part of the message dictionary.

Editing Common Messages

Because a common message can be used in any application, consider the ramifications if you edit any aspect of the message, including incident and logging settings. Changes would be reflected in all instances where the message is used. For example, if you change the message text, make sure that the text

would make sense to all users across Oracle Fusion Applications who might see it.

Creating Common Messages

You can create custom common messages for use in multiple places within a single product. Do not begin the message name with FND_CMN, but use another suitable convention. The message number should be within the range that is designated for the product.

Define 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 sets of books, entering transactions, and recording balances. Only USD is enabled by default.

What's the difference between precision, extended precision, and minimum accountable unit for a currency?

Precision is the number of digits to the right of the decimal point used in regular currency transactions. Extended precision is the number of digits to the right of the decimal point used in calculations for this currency, and it must be greater than or equal to the standard precision. For example, USD would have 2 for precision because amounts are transacted as such, for example \$1.00. For calculations, for example adding USD amounts, you might want the application to be more precise than two decimal digits, and would enter an extended precision accordingly.

Note

Some applications use extended precision. Others, such as Oracle Fusion General Ledger, do not.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for the cent. This unit does not necessarily correspond to the precision for all currencies.

What's a statistical unit currency type?

The statistical unit currency type is used only for the Statistical (STAT) currency. The Statistical currency is used to record statistics such as the number of items bought and sold. Statistical balances can be used directly in financial reports, allocation formulas, and other calculations.

When do I create or edit ISO languages?

You can edit the names and descriptions of International Organization for Standardization (ISO) languages to determine how they are displayed in lists of ISO language values in Oracle Fusion Applications. The ISO languages are from the ISO 639 standard. If there were changes to the ISO standard and the system has not yet been patched with the latest ISO values, you can update the ISO alpha-2 code or add languages as needed.

When do I edit languages?

Installed languages automatically appear on the Manage Languages page, so you do not manually enter newly installed languages. This page contains all languages available for installation and translation in Oracle Fusion Applications. Each dialect is treated as a separate language. The language codes and names are values used by the system.

You generally would not edit any of the detailed fields unless you really need to and know what they are.

When do I create or edit time zones?

Though all standard time zones are provided, optionally enable only a subset for use in lists of time zone values in Oracle Fusion Applications. You can add time zones if new zones became standard and the system has not yet been patched with the latest values.

Manage Data Security Policies

Data Security in the Security Reference Implementation: Explained

The reference implementation contains a set of data security policies that can be inspected and confirmed to be suitable or a basis for further implementation using the Authorization Policy Manager (APM).

The security implementation of an enterprise is likely a subset of the reference implementation, with the enterprise specifics of duty roles, data security policies, and HCM security profiles provided by the enterprise.

The business objects registered as secure in the reference implementation are database tables and views.

Granting or revoking object entitlement to a particular user or group of users on an object instance or set of instances extends the base Oracle Fusion Applications security reference implementation without requiring customization of the applications that access the data.

Data Security Policies in the Security Reference Implementation

The data security policies in the reference implementation entitle the grantee (a role) to access instance sets of data based on SQL predicates in a WHERE clause.

Tip

When extending the reference implementation with additional data security policies, identify instance sets of data representing the business objects that need to be secured, rather than specific instances or all instances of the business objects.

Predefined data security policies are stored in the data security policy store, managed in the Authorization Policy Manager (APM), and described in the Oracle Fusion Applications Security Reference Manual for each offering. A data security policy for a duty role describes an entitlement granted to any job role that includes that duty role.

Warning

Review but do not modify HCM data security policies in APM except as a custom implementation. Use the HCM Manage Data Role And Security Profiles task to generate the necessary data security policies and data roles.

The reference implementation only enforces a portion of the data security policies in business intelligence that is considered most critical to risk management without negatively affecting performance. For performance reasons it is not practical to secure every level in every dimension. Your enterprise may have a different risk tolerance than assumed by the security reference implementation.

HCM Security Profiles in the Security Reference Implementation

The security reference implementation includes some predefined HCM security profiles for initial usability. For example, a predefined HCM security profile allows line managers to see the people that report to them.

The IT security manager uses HCM security profiles to define the sets of HCM data that can be accessed by the roles that are provisioned to users

Data Roles

The security reference implementation includes no predefined data roles to ensure a fully secured initial Oracle Fusion Applications environment.

The security reference implementation includes data role templates that you can use to generate a set of data roles with entitlement to perform predefined business functions within data dimensions such as business unit. Oracle Fusion Payables invoicing and expense management are examples of predefined business functions. Accounts Payable Manager - US is a data role you might generate from a predefined data role template for payables invoicing if you set up a business unit called US.

HCM provides a mechanism for generating HCM related data roles.

Data Security: Explained

By default, users are denied access to all data.

Data security makes data available to users by the following means.

- Policies that define grants available through provisioned roles
- Policies defined in application code

You secure data by provisioning roles that provide the necessary access. Enterprise roles provide access to data through data security policies defined for the inherited application roles.

When setting up the enterprise with structures such as business units, data roles are automatically generated that inherit job roles based on data role templates. Data roles also can be generated based on HCM security profiles. Data role templates and HCM security profiles enable defining the instance sets specified in data security policies.

When you provision a job role to a user, the job role implicitly limits data access based on the data security policies of the inherited duty roles. When you provision a data role to a user, the data role explicitly limits the data access of the inherited job role to a dimension of data.

Data security consists of privileges conditionally granted to a role and used to control access to the data. A privilege is a single, real world action on a single business object. A data security policy is a grant of a set of privileges to a principal on an object or attribute group for a given condition. A grant authorizes a role, the grantee, to actions on a set of database resources. A database resource is an object, object instance, or object instance set. An entitlement is one or more allowable actions applied to a set of database resources.

Data is secured by the following means.

Data security feature	Does what?
Data security policy	Grants access to roles by means of entitlement
Role	Applies data security policies with conditions to users through role provisioning.
Data role template	Defines the data roles generated based on enterprise setup of data dimensions such as business unit.
HCM security profile	Defines data security conditions on instances of object types such as person records, positions, and document types without requiring users to enter SQL code
Masking	Hides private data on non-production database instances
Encryption	Scrambles data to prevent users without decryption authorization from reading secured data

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.

Who	can do	what	on which set of data
Accounts payable managers	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.

Oracle Fusion Payroll uses HCM security profiles to secure project organizations. Applications outside of HCM can use the HCM Data Roles UI pages to give their roles access to HR people.

Masking and Encryption

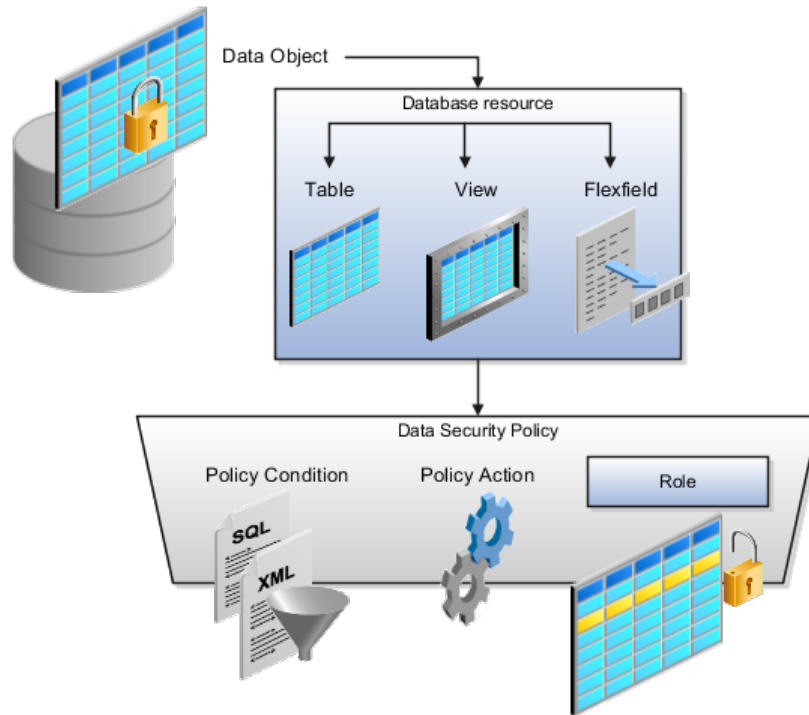
Oracle Fusion Applications uses masking to protect sensitive data from view by unauthorized users. Encryption APIs mask sensitive fields in applications user interfaces. Additionally, Oracle Data Masking is available for masking data in non-production instances and Oracle Transparent Data Encryption is available for protecting data in transit or in backups independent of managing encryption keys.

Database Resources and Data Security Policies: How They Work Together

A data security policy applies a condition and allowable actions to a database resource for a role. When that role is provisioned to a user, the user has access to data defined by the policy. In the case of the predefined security reference implementation, this role is always a duty role. Data roles generated to inherit the job role based on data role templates limit access to database resources in a particular dimension, such as the US business unit.

The database resource defines and instance of a data object. The data object is a table, view, or flexfield.

The following figure shows the database resource definition as the means by which a data security policy secures a data object. The database resource names the data object. The data security policy grants to a role access to that database resource based on the policy's action and condition.



Database Resources

A database resource specifies access to a table, view, or flexfield that is secured by a data security policy.

- Name providing a means of identifying the database resource
- Data object to which the database resource points

Data Security Policies

Data security policies consist of actions and conditions for accessing all, some, or a single row of a database resource.

- Condition identifying the instance set of values in the data object
- Action specifying the type of access allowed on the available values

Note

If the data security policy needs to be less restrictive than any available database resource for a data object, define a new data security policy.

Actions

Actions correspond to privileges that entitle kinds of access to objects, such as view, edit, or delete. The actions allowed by a data security policy include all or a subset of the actions that exist for the database resource.

Conditions

A condition is either a SQL predicate or an XML filter. A condition expresses the values in the data object by a search operator or a relationship in a tree hierarchy. A SQL predicate, unlike an XML filter, is entered in a text field in the data security user interface pages and supports more complex filtering than an XML filter, such as nesting of conditions or sub queries. An XML filter, unlike a SQL predicate, is assembled from choices in the UI pages as an AND statement.

Tip

An XML filter can be effective in downstream processes such as business intelligence metrics. A SQL predicate cannot be used in downstream metrics.

Securing Data Access: Points to Consider

Oracle Fusion Applications supports securing data through role-based access control (RBAC) by the following methods.

Method of securing data	Reason	Example
Data roles apply explicit data security policies on job and abstract roles	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.	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.
Data security policies	Define data access for application roles and provide inheriting job and abstract roles with implicit data security	Projects

If a user has access to the same function through different roles that access different data sets, then the user has access to a union of those data sets.

When a runtime session is created, Oracle Platform Security Services (OPSS) propagates only the necessary user to role mapping based on Oracle Fusion Data Security grants. A grant can specify entitlement to the following.

- Specific rows of data (data object) identified by primary key
- Groups of data (instance set) based on a predicate that names a particular parameter
- Data objects or instance sets based on runtime user session variables

Data is either identified by the primary key value of the row in the table where the data is stored. Or data is identified by a rule (SQL predicate) applied to the WHERE clause of a query against the table where the data is stored.

Grants

Oracle Fusion Data Security can be used to restrict the following.

- Rows that are returned by a given query based on the intended business operation
- Actions that are available for a given row

Grants control which data a user can access.

Note

Attribute level security using grants requires a data security policy to secure the attribute and the entitlement check enforces that policy.

A grant logically joins a user or role and an entitlement with a static or parameterized object instance set. For example, `REGION='WEST'` is a static object instance set and `REGION=&GRANT_ALIAS.PARAMETER1` is a parameterized object instance set. In the context of a specific object instance, grants specify the allowable actions on the set of accessible object instances. In the database, grants are stored in `FND_GRANTS` and object instance sets are stored in `FND_OBJECT_INSTANCE_SETS`. Object access can be tested using the privilege check application programming interface (API).

Securing a Business Object

A business object is a logical entity that is typically implemented as a table or view, and corresponds to a physical database resource. The data security policies of the security reference implementation secure predefined database resources. Use the Manage Data Security Policies task to define and register other database resources.

Data security policies identify sets of data on the registered business object and the actions that may be performed on the business object by a role. The grant can be made by data instance, instance set or at a global level..

Note

Use parameterized object instance sets whenever feasible to reduce the number of predicates the database parses and the number of administrative intervention required as static object instances sets become obsolete. In HCM, security profiles generate the instance sets.

Data Role Templates: Explained

You use data role templates to generate data roles. You generate such data roles, and create and maintain data role templates in the Authorization Policy Manager (APM).

Note

HCM data roles are generated using the Manage Data Roles and Security Profiles task, which uses HCM security profiles, not data role templates, to define the data security condition.

The following attributes define a data role template.

- Template name
- Template description
- Template group ID
- Base roles
- Data dimension
- Data role naming rule
- Data security policies

The data role template specifies which base roles to combine with which dimension values for a set of data security policies. The base roles are the parent job or abstract roles of the data roles.

Note

Abstract, job, and data roles are enterprise roles in Oracle Fusion Applications. Oracle Fusion Middleware products such as Oracle Identity Manager (OIM) and Authorization Policy Manager (APM) refer to enterprise roles as external roles. Duty roles are implemented as application roles in APM and scoped to individual Oracle Fusion Applications.

The dimension expresses stripes of data, such as territorial or geographic information you use to partition enterprise data. For example, business units are a type of dimension, and the values picked up for that dimension by the data role template as it creates data roles are the business units defined for your enterprise. The data role template constrains the generated data roles with grants of entitlement to access specific data resources with particular actions. The data role provides provisioned users with access to a dimensional subset of the data granted by a data security policy.

An example of a dimension is a business unit. An example of a dimension value is a specific business unit defined in your enterprise, such as US. An example of a data security policy is a grant to access a business object such as an invoice with a view entitlement.

When you generate data roles, the template applies the values of the dimension and participant data security policies to the group of base roles.

The template generates the data roles using a naming convention specified by the template's naming rule. The generated data roles are stored in the Lightweight Directory Access Protocol (LDAP) store. Once a data role is generated, you provision it to users. A user provisioned with a data role is granted permission to access the data defined by the dimension and data security grant policies of the data role template.

For example, a data role template contains an Accounts Payable Specialist role and an Accounts Payable Manager role as its base roles, and region as its dimension, with the dimension values US and UK. The naming convention is [base-role-name]:[DIMENSION-CODE-NAME]. This data role template generates four data roles.

- Accounts Payable Specialist - US (business unit)
- Accounts Payable Specialist - UK (business unit)
- Accounts Payable Manager - US (business unit)
- Accounts Payable Manager - UK (business unit)

Making Changes To Data Role Templates

If you add a base role to an existing data role template, you can generate a new set of data roles. If the naming rule is unchanged, existing data roles are overwritten.

If you remove a base role from a data role template and regenerate data roles, a resulting invalid role list gives you the option to delete or disable the data roles that would be changed by that removal.

Making Changes to Dimension Values

If you add a dimension value to your enterprise that is used by a data role template, you must regenerate roles from that data role template to create a data role for the new dimension. For example if you add a business unit to your enterprise, you must regenerate data roles from the data role templates that include business unit as a dimension.

If you add or remove a dimension value from your enterprise that is used to generate data roles, regenerating the set of data roles adds or removes the data roles for those dimension values. If your enterprise has scheduled regeneration as an Oracle Enterprise Scheduler Services process, the changes are made automatically.

For information on working with data role templates, see the Oracle Fusion Middleware Administrator's Guide for Authorization Policy Manager (Oracle Fusion Applications Edition).

Define Document Sequences

Document Sequences: Explained

In Oracle Fusion Applications, each business document or business event is uniquely identified by a document sequence number that you assign to it. However, the document sequencing feature must be turned on (enabled) on the business document or event to allow the assignment. For example, if document sequencing is enabled, you can assign a document sequence number

to an invoice that gets generated in response to a purchase order. You can use document sequences as a proof to track successfully executed transactions as well as failed transactions. Additionally, a document sequence helps in generating an audit trail, which can be used to identify how a particular transaction passed through various applications.

Document sequencing can be managed automatically, manually, and gaplessly.

Note

Plan your document sequencing carefully before you use the options available in the application to apply sequence numbers. Avoid changes to the options after you saved your work on the Manage Document Sequences and Manage Document Sequence Categories pages.

Automatic Sequencing

Automatic document sequencing assigns a unique number to each document as it is generated, and this unique number is stored in the database. The numbering is sequential by date and time of creation. If you define a sequence to automatically number documents, you can provide an initial value to begin the sequence. In absence of a custom value, the default value 1 is used.

Manual Sequencing

Manual sequencing requires you to assign a unique number to each document before it is generated. In manual sequencing, the numerical ordering and completeness of a transaction is not enforced. Users can skip or omit numbers when entering the sequence value. However, each time that a number is assigned, the application validates its uniqueness.

Gapless Sequencing

Gapless sequencing is similar to automatic sequencing. It automatically generates a unique number for each document, but does that only for successfully generated documents. As a result, the sequence is maintained for all the documents that are generated, and no sequence numbers are lost due to incomplete or failed document generation.

Important

Use this type of sequencing only if necessary because it may affect the performance of the system and slow down transaction processing.

Document Sequence Categories: Explained

A document sequence category is a set of documents that share similar characteristics and that are formed into a logical group. Document sequence categories simplify the task of assigning number sequences to specific documents. Instead of assigning a number to each document, you assign a document sequence to one or more document sequence categories. The document sequence category automatically takes care of numbering the documents.

A document sequence category identifies the database table that stores documents resulting from transactions that your users enter. When you assign a sequence to a category, the sequence numbers the documents that are stored in a particular table. You must create document sequence categories to be able to manage the task of assigning document sequences.

Restriction

Once a document sequence category is created, you cannot change the application, the category code, or the table name. Therefore, carefully consider these details and plan your document sequencing requirement before you begin working with the application.

Once you create a document sequence category, it is available for use under the **Document Sequences: Assignments** section on the Manage Document Sequences page. The **Category** field contains the name of the document sequence category. After you create a document sequence, you can assign it to a document sequence category.

Document Sequences: Points to Consider

Sequencing documents is an important business and legal requirement. Certain aspects of the defining process are permanent and cannot be modified later. Therefore, it is important that you first decide the appropriate document sequence to use for a set of documents. You must also decide beforehand the type of document sequencing, because you are not allowed to switch to other types once a sequence is assigned to a document sequence category. Make a note of the details such as the document sequence and document sequence category so that you can refer to them at a later point in time. Also note if there are any restrictions or configuration prerequisites before you define document sequencing.

Note

Products that implement document sequencing have specifications about its usage. Refer to the corresponding product documentation for specific details and also to determine if there are any restrictions or configuration prerequisites.

Creating and Editing Document Sequences

You can create document sequences that are automatic, manual, or gapless, depending on the business or legal requirement. By default, the current date is considered as the start date. If the end date is left blank, it means that the sequence definition never expires. Among the several options used in creating and editing document sequences, the following options are functionally more important and therefore need to be carefully determined:

- **Determinant Type:** Select to limit the document sequencing activity to certain documents that belong to a specific business entity, such as Ledger, Tax Registration, and so on.

- **Initial Value:** Enter a value for the first document in your sequence. This field applies only to sequences with automatic or gapless numbering types. Sequence numbers should not be greater than eight digits. If you leave this field blank, the first document is automatically assigned a value of 1. Once a document sequence is defined, you cannot change this initial value.

Creating and Editing Document Sequence Categories

Document sequence categories are defined to make it easy to assign document sequence definitions to a group of documents instead of to individual documents. Each document sequence category is mapped to a specific table, where the documents belonging to that category are stored. The table must already be enabled for document sequencing. When specifying the table, you must consider the following points:

- When the sequential numbering feature checks for completeness or generates a report, it locates the category's documents in the table.
- You can select only tables belonging to the application associated with the category.
- Once a category is defined, you cannot change the choice of table.

Assigning Document Sequences

Identify the documents to be numbered before assigning them a document sequence. For each document sequence, there can be only one active assignment to a document sequence category, a method code, and a determinant value (if applicable). As part of the assignment, specify whether the document is created automatically (for example, due to a batch process, or manually through a form). If you do not specify an end date, the assignment continues to remain active throughout the process cycle. If a determinant type was specified for the document sequence, then enter a specific determinant value related to the selected determinant type.

At runtime, when users create documents, the document sequence to be assigned is determined by finding the active assignment that matches the correct combination of category, numbering method, and the date range containing the transaction date.

Auditing Document Sequences

You can audit document sequences, if required, to provide an audit trail of the document sequences used in a specific product. However, before enabling the audit functionality for a document sequence, you must have created an audit table for the specific document sequence, using appropriate details. Enabling the audit functionality is permitted only for newly created document sequences. You cannot audit document sequences that are already in use by a specific product.

For more information about defining a document sequence audit table, see the Oracle Fusion Applications Developer's Guide.

Manage Public Unique Identifier Profile Options

Public Unique Identifier: Explained

Oracle Fusion applications generate thousands of rows of data that require unique identifiers (UIDs). These automatically generated, 18-digit numbers are not easily read or used by humans. A cogent example is an electronic airline ticket whose numbers can be 13 digits long, such as, 0162128736572. There are also confirmation or itinerary numbers that relate directly to that ticket that are only six digits long, such as QDLG9S; these can be used interchangeably with the ticket number. These shorter confirmation numbers are much easier to read and process by passengers and employees alike. A separate Public Unique ID (PUID) is also generated by the document sequencing feature and is available for encoding into a more user-friendly, alphanumeric ID.

The Oracle Fusion default setup shows the automatically generated numeric private UIDs and PUIDs for data rows. To use an alphanumeric PUID instead of the delivered numeric, navigate to the Setup and Maintenance task **Manage Public Unique Identifier Profile Options** to enable the encoding process. There are two profile options to configure the PUID:

- **CRM Public Unique ID String Encoding** controls the characters used in the encoding of the PUID based on a radix, or base number.
- **CRM Public Unique ID Prefix** defines the optional prefix value for the PUID.

Document sequencing typically begins with the number one. You may choose to start the sequences of your PUIDs at a specific value. See the related links below to determine the document sequencing.

CRM Public Unique ID String Encoding Profile Option

This profile option determines the set of numbers and letters to be used in encoding the PUID generated by the document sequencing feature. The PUID is delivered with the encoding set to none. To enable the encoding of the PUID, choose a radix, or base conversion algorithm. The available values are stored in the lookup type **ZCA_PUID_ENCODING**. These encoding methods convert the PUID into user readable IDs using alphanumeric characters rather than just numeric digits. For example, if **BASE_26** is chosen, only the upper case letters of the alphabet will be used.

There are seven delivered base values. These are not extensible.

RADIX	Values
BASE_16	Numbers 0-9, letters A-F
BASE_26	Letters A-Z, Upper Case
BASE_29	Numbers 2-9, letters BCDFGHJKLMNPQRSTUVWXYZ
BASE_31	Numbers 0-9, letters ABCDEFGHIJKLMNPQRTUWXY
BASE_36	Numbers 0-9, letters A-Z upper case
BASE_62	Numbers 0-9, letters A-Z, letters a-z

NONE (default)	No Encoding
----------------	-------------

CRM Public Unique ID Prefix Profile Option

A prefix may be defined as a site-level profile option. This profile option is delivered with no prefix defined. When defined, the inserted text is prepended to the PUID base encoded document sequence value. For example, if the PUID was CLE123 and the Prefix was ORA, the new PUID would be ORACLE123. Any delimiters or separation characters desired must be defined in the prefix itself. Ensure that the prefix length is short enough so that the concatenated PUID and prefix do not exceed the defined field length, usually no more than 30 characters. For example, you may want the records for the pharmaceutical divisions of your company to be denoted with PUIDs and the prefix Pharma- or Pharma1, Pharma2, and so on.

Define Trees

Trees: Overview

Use the tree management feature in Oracle Fusion applications to organize data into hierarchies. A hierarchy contains organized data and enables the creation of groups and rollups of information that exist within an organization. Trees are hierarchical structures that enable several data management functions such as better access control, application of business rules at various levels of hierarchies, improved query performance, and so on.

For example, XYZ Corporation has two departments: Marketing and Finance. The Finance department has two functional divisions: Receivables and Payables. Defining a tree for the XYZ Corporation establishes a hierarchy between the organization and its departments, and between the departments and their respective functional divisions. Such a hierarchical modeling of organizational data could be used for executing several data management functions within that organization.

You can create one or more versions of trees, and they can be labeled for better accessibility and information retrieval. You can create trees for multiple data sources, which allow the trees to be shared across Oracle Fusion applications.

Tree Structures

A tree structure is a representation of the data hierarchy, and guides the creation of a tree. A tree is an instance of the hierarchy as defined in the tree structure. Tree structures enable you to enforce business rules to which the data must adhere.

The root node is the topmost node of a tree. Child nodes report to the root node. Child nodes at the same level, which report to a common parent node, are called siblings. Leaves are details branching off from a node but not extending further down the tree hierarchy.

Tree Versions

A tree is created having only one version. However, users can create more than one tree version depending on the need, and they can make changes to those

versions. Depending on varying requirements, users can create one or more tree versions and publish all of them or some of them by making the versions active at the same time. Similar to any other version control system, versions of trees are maintained to keep track of all the changes that a tree undergoes in its life cycle.

Tree Labels

Tree labels are short names associated with trees and tree structures and point directly to the data source. Tree labels are automatically assigned to the tree nodes. You can store labels in any table and register the label data source with the tree structure.

Manage Tree Structures

Tree Structures: Explained

A tree structure defines the hierarchy for creating trees and prescribes rules based on which trees are created, versioned, and accessed. You can associate multiple data sources with a tree structure. A tree is an instance of this hierarchy. Every tree structure can contain one or more trees.

You can create tree structures specific to an application but you can share tree structures across applications. If you apply version control to the tree structure, it is carried over to the trees that are based on the tree structure. Each tree version contains at least one root node. Occasionally, a tree version may have more than one root node.

An administrator controls the access to tree structures through a set of rules that are periodically audited for validity.

Tree Structure Definition: Points to Consider

Defining a tree structure involves specifying several important pieces of information on the **Create Tree Structure: Specify Definition** page.

Tree Node Selection

The **Tree Node** table displays data in nodes that exist in the data hierarchy. You must select the correct and most appropriate tree node table to be able to define the tree structure, based on the tree hierarchy you want to establish. This selection also affects the level of security that is set on a tree node and its child entities.

Tree Sharing Mode

The following options are used to determine the mode of sharing a tree structure across the applications.

- **Open:** Indicates that the tree is associated with all reference data sets.
- **Set ID:** Indicates that the tree will be associated with a specific reference data set.

Creation Mode

Indicates the source where the tree structure is being defined. For predefined tree structures select Oracle and for custom structures, select Customers.

Customization

You can customize the predefined tree structures as well as the ones that you created. However, customizing the predefined tree structures involves certain level of access restrictions, and will be limited to specific tree nodes and downwards in hierarchy.

Multiple Tree Versions

One or more trees and tree versions can be based on a tree structure. A tree structure can have one or more trees and tree versions based on it. Usually, only one active version is permitted at any given point of time. However, depending on the requirement, you can allow two or more tree versions to be in the active state for the same date range. This flexibility allows you to choose the tree version that you want to implement.

Managing Tree Structures: Points to Consider

You can create, edit, and delete tree structures depending upon the requirement. You can also audit and change the status a tree structure.

Creating and Editing Tree Structures

You can create trees on the basis of a tree structure. When you edit an active tree structure, the status of the tree structure and all associated trees and their versions change to draft. To reuse a tree structure, you can create a copy of it without copying the associated trees and tree versions. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.

Note

For specific information on working with the predefined tree structures that exist in an Oracle Fusion application, refer to the specific product documentation.

Setting Status

If you change the status of a tree structure, the status of the trees and tree versions associated with that tree structure also changes.

The following table lists the different statuses of a tree structure.

Status	Meaning
Draft	Yet to be published or is in a modified state.

Active	In use and based on which one or more trees or tree versions are created.
Inactive	Not in use.

Tree Structure Audit Results: Explained

Use the tree structure audit results to verify the tree structure's correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- The result of the validation, including a detailed message
- Corrective actions to take if there are any validation errors

Running an Audit

Setting the status of a tree structure to active automatically triggers an audit of that tree structure. You can also manually trigger an audit on the manage Tree Structures page, using **Actions - Audit**. The Tree Structure Audit Result table shows a list of validations that ran against the selected tree structure.

Validation Details

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
Restrict By Set ID	On the Manage Tree Structures: Specify Data Sources page, if the Set ID check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have a reference data set attribute. This validation does not take place when the check box is not selected.	Even when the check box is selected, one or more of its data source view objects does not contain a reference data set attribute.	If reference data set restriction is required for this tree structure, include a reference data set attribute on all data sources. Otherwise, deselect the check box.
Row Flattened Table Name	On the Manage Tree Structures: Specify Performance Options page, a valid row flattened table must be specified for the tree structure. It can either be the standard row flattened table <code>FND_TREE_NODE_RF</code> or a custom table.	<ul style="list-style-type: none"> • The specified table does not exist in the database. • The specified table does not contain the same columns as the <code>FND_TREE_NODE_RF</code> table. 	Correct the row flattened table definition.

<p>Available Label Data Sources</p>	<p>On the Manage Tree Structures: Specify Data Sources page, if a labeling scheme is specified for the tree structure by selecting a list item from the Labeling Scheme list box, the label data source view object specified for each data source must be accessible, and the primary keys must be valid. This restriction does not apply when you select None from the Labeling Scheme list box.</p>	<ul style="list-style-type: none"> • Any of the specified label data source view objects do not exist. • Any of the specified label data source view objects do not have primary keys. • When a label data source view object is initially defined, the database registers the primary keys for the view object. If the view object is later modified such that its primary keys no longer match the primary keys that were registered earlier, this validation fails. 	<ul style="list-style-type: none"> • Correct the specified label data source view object. • Correct the primary keys of the specified label data source view object. • Either correct the primary keys in the label data source view object to match the primary keys that were earlier registered in <code>FND_TS_DATA_SOURCE</code>, or correct the primary keys registered in that table to match the new view object definition.
-------------------------------------	---	---	---

Available Data Sources	Each data source view object specified for the tree structure must be accessible, and all its primary key attributes must be valid.	<ul style="list-style-type: none"> • Any of the specified data source view objects do not exist. • When a data source view object is initially defined, the database automatically registers the primary keys for the view object if the Use non-defined primary key columns check box on the Data Source dialog box is not selected. If the check box is selected, the database registers the primary keys specified explicitly by the user on the Add Data Source dialog box. If the registered primary keys contain any duplicates, this validation fails. • The Use non defined primary key columns check box is selected in a data source, but the list of specified primary key columns does not match the primary keys defined in the corresponding data source view object. • Any common attribute that exists in both the data source view object and the tree node view object is not of the same data type in both view objects. 	<ul style="list-style-type: none"> • Correct the specified data source view object. • Correct the duplicate column in the registered primary keys. • Correct the primary keys of the specified data source view object. • Correct any mismatch in data types.
------------------------	---	---	---

Column Flattened Table Name	On the Manage Tree Structures: Specify Performance Options page, a valid column flattened table must be specified for the tree structure. It can either be the standard row flattened table <code>FND_TREE_NODE_CF</code> or a custom table.	<ul style="list-style-type: none"> The specified table does not exist in the database. The specified table does not contain the same columns as the <code>FND_TREE_NODE_CF</code> table. 	Correct the column flattened table definition.
Restrict by Date	On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have effective start date and effective end date attributes. This validation does not take place when the check box is not selected.	Even when the check box is selected, one or more of its data source view objects does not contain effective start date and effective end date attributes.	If the date restriction is required for this tree structure, include the effective start date and effective end date attributes on all data sources. Otherwise, deselect the check box.
Tree Node Table Name	On the Manage Tree Structures: Specify Definition page, a valid tree node table must be specified for the tree structure. It can either be the standard row flattened table <code>FND_TREE_NODE</code> or a custom table.	<ul style="list-style-type: none"> No table is specified in the Tree Node Table field. The specified table does not exist in the database. The specified table does not contain the same columns as the <code>FND_TREE_NODE</code> table. 	Correct the tree node table definition.
Allow Node Level Security	If the Allow Node Level Security option is set to No for the tree structure, the same option cannot be set to Yes on any of its data sources. This is a database setting that is not visible on the Manage Tree Structures page.	The option is set to No for the tree structure but one or more associated data sources have that option set to Yes.	Correct the option setting in the tree structure and their data sources.

Specifying Data Sources for Tree Structures: Points to Consider

The data sources provide the items for establishing hierarchy in a tree structure. In the tree management infrastructure, these data sources are Oracle Application

Development Framework (ADF) business components view objects, which are defined by application development.

Labeling Schemes

Selecting a labeling scheme determines how the tree nodes are labeled. You may select a labeling scheme to assign at the data source level, at the parent node level, or keep it open for customer assignment. You may also choose not to have any labeling scheme. However, if you decide to use any of the labeling schemes, you may need to select the following additional options, to restrict the list of values that appear under the selected tree node.

- **Allow Ragged Nodes:** To include nodes that have no child nodes, and are shorter than the remaining nodes in the entire hierarchy.
- **Allow Skip Level Nodes:** To include nodes that are at the same level but have parent nodes at different levels.

Restriction of Tree Node Values

You can decide the depth of the tree structure by selecting an appropriate value from the list. Keeping the depth limit open renders an infinite list of values. Using the following options, you can restrict the list of values that appear for selection under a specific tree node.

- **Date Range:** Specifies whether a selection of nodes should be restricted to the same date range as the tree version.
- **Allow Multiple Root Nodes:** Allows you to add multiple root nodes when creating a tree version.
- **Reference Data Set:** Specifies whether a selection of nodes should be restricted to the same set as the tree.

Data Source Values and Parameters

Tree data sources have optional data source parameters with defined view criteria and associated bind variables. You can specify view criteria as a data source parameter when creating a tree structure, and edit the parameters when creating a tree. Multiple data sources can be associated with a tree structure and can have well-defined relationships among them.

Note

Parameter values customized at the tree level override the default values specified at the tree-structure level.

The data source parameters are applied to any tree version belonging to that data source, when performing node operations on the tree nodes. Data source parameters also provide an additional level of filtering for different tree structures. The tree structure definition supports three data source parameter types.

- **Bound Value:** Captures any fixed value, which is used as part of the view criteria condition.
- **Variable:** Captures and binds a dynamic value that is being used by the data source view object. This value is used by the WHERE condition of the data flow.
- **View Criteria:** Captures the view criteria name, which is applied to the data source view object.

You can also specify which of the data source parameters are mandatory while creating or editing the tree structure.

View objects from the ADF business components are used as data sources. To associate the view object with the tree structure, you can pick the code from ADF business component view objects and provide the fully qualified name of the view object, for example, oracle.apps.fnd.applcore.trees.model.view.FndLabelVO.

Specifying Performance Options for a Tree Structure: Points to Consider

Tree structures are heavily loaded with data. As a tree management guideline, use the following settings to improve performance of data rendering and retrieval.

- Row Flattening
- Column Flattening
- Column Flattened Entity Objects
- ADF Business Component View Objects

Row Flattening

Row flattening optimizes parent-child information for run-time performance by storing additional rows in a table for instantly finding all descendants of a parent without initiating a CONNECT BY query. Row flattening eliminates recursive queries, which allows operations to perform across an entire subtree more efficiently.

To store row flattened data for the specific tree structure, users can either use the central `FND_TREE_NODE_RF` table or they can register their own row flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a row-flattened table contains an additional row with Corporation directly being the parent of Region (Corporation-Region).

Column Flattening

Column flattening optimizes parent-child information for run-time performance by storing an additional column in a table for all parents of a child.

To store column flattened data for the specific tree structure, users can either use the central `FND_TREE_NODE_CF` table or they can register their own column flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a flattened table in addition to these columns, contains three new columns: Region, Sales Division, and Corporation. Although positioned next to each other, the column Region functions at the lower level and Corporation at the higher level, retaining the data hierarchy.

Column Flattened Entity Objects

In the absence of a column-flattened table, if you need to generate the business component view objects for your tree structure for the flattened table, use the tree management infrastructure to correctly provide the fully qualified name of the entity object for the column flattened table.

ADF Business Component View Objects

View objects from the ADF business components can also be used as data sources, eliminating the need to create new types of data sources. This field is to store the fully qualified name for the business component view object generated by the tree management for business intelligence reporting and usage. The business component view object is a combination of the tree data source and column flattened entity. Using this option prevents data redundancy and promotes greater reuse of existing data, thereby improving the performance of the tree structure.

Manage Tree Labels

Tree Labels: Explained

Tree labels are tags that are stored on tree nodes. You can store labels in any table and register the label data source with the tree structure. When a labeling scheme is used for trees, the selected labels are stored in the tree label entity and each tree node contains a reference to a tree label in the labeling scheme.

The following table lists the three ways in which tree labels are assigned to the tree nodes.

Labeling Scheme	Description
Level	Labels that are automatically assigned based on the data source to which the tree node belongs. A level label points to a specific data source. For example, in a tree that reflects the organizational hierarchy of an enterprise, all division nodes appear on one level and all department nodes on another.
Group	Labels that you can arbitrarily assign to tree nodes.
Depth	Labels that are automatically assigned based on the depth of the tree node within the tree. No manual assignment is performed. <hr/> Note In an unbalanced hierarchy, a level may not be equal to depth.

Manage Trees and Tree Versions

Managing Trees and Tree Versions: Points to Consider

You can create and edit trees and tree versions depending upon the requirement. A tree can have one or more tree versions. Typically, when changes are made to an existing tree, a new version is created and published.

Creating and Editing Trees

Trees are created based on the structure defined in the tree structure. You can create trees, modify existing trees, and delete trees. If you want to copy an existing tree, you can duplicate it. However, only the tree is duplicated and not its versions.

Creating a tree involves specifying the tree definition and specifying the labels that are used on its nodes. If the selected tree structure has data sources and parameters defined for it, they appear on the page allowing you to edit the parameter values at the tree node level.

Note

Parameter values customized at the tree level will override the default values specified at the tree-structure level.

Creating and Editing Tree Versions

Tree versions are created at the time of creating trees. A tree must contain a version.

Editing an existing tree provides you the choice to update the existing version. You can also edit the existing version that lies nested under the tree in the search results.

When you edit a tree version bearing Active status, the status changes to Draft until the modifications are saved or cancelled.

Tree Version Audit Results: Explained

Use the tree version audit results to verify the tree version's correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- The result of the validation, including a detailed message
- Corrective actions to take if there are any validation errors

Running an Audit

An audit automatically runs whenever a tree version is set to active. You can also manually trigger an audit on the Manage Trees and Tree Versions page, using **Actions - Audit**. The Tree Version Audit Result table shows a list of validations that ran against the selected tree version.

Validation Details

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
Effective Date	The effective start and end dates of the tree version must be valid.	The effective end date is set to a value that is not greater than the effective start date.	Modify the effective start and end dates such that the effective start date is earlier than the effective end date.
Root Node	On the Manage Tree Structures: Specify Data Sources page, if the Allow Multiple Root Nodes check box for the Restrict Tree Node List of Values Based on option is not selected, and if the tree structure is not empty, the tree version must contain exactly one root node. This validation does not take place if the check box is selected.	Even if the check box is deselected, the tree version has multiple root nodes.	Modify the tree version such that there is exactly one root node.
Data Source Max Depth	For each data source in the tree structure, on the Data Source dialog box, if the data source is depth-limited, the data in the tree version must adhere to the specified depth limit. This validation does not apply to data sources for which the Maximum Depth field is set to Unlimited .	The tree version has data at a depth greater than the specified depth limit on one or more data sources.	Modify the tree version such that all nodes are at a depth that complies with the data source depth limit.
Duplicate Node	On the Data Source dialog box, if the Allow Duplicates check box is not selected, the tree version should not contain more than one node with the same primary key from the data source. If the check box is selected, duplicate nodes are permitted.	Even when the check box is deselected, the tree version contains duplicate nodes.	Remove any duplicate nodes from the tree version.

Available Node	All nodes in the tree version should be valid and available in the underlying data source.	<ul style="list-style-type: none"> • A node in the tree version does not exist in the data source. Deleting data items from the data source without removing the corresponding nodes from the tree version can result in orphaned nodes in the tree version. For example, if you added node A into your tree version, and subsequently deleted node A from the data source without removing it from the tree version, the validation fails. • The tree version contains a tree reference node, which references another tree version that does not exist. 	Remove any orphaned nodes from the tree version. Update tree reference nodes so that they reference existing tree versions.
Node Relationship	All nodes must adhere to the relationships mandated by the data sources registered in the tree structure.	The tree structure has data sources arranged in a parent-child relationship, but the nodes in the tree do not adhere to the same parent-child relationship. For example, if the tree structure has a Project data source with a Task data source as its child, Task nodes should always be under Project nodes in the tree version. This validation fails if there are instances where a Project node is added as the child of a Task node.	Modify the tree version such that the nodes adhere to the same parent-child relationships as the data sources.

SetID Restricted Node	On the Manage Tree Structures: Specify Data sources page, if the Set ID check box is selected to enable the Restrict Tree Node List of Values Based on option for each tree node, the underlying node in the data source must belong to the same reference data set as the tree itself. This restriction does not apply when the check box is not selected.	Even when the check box is selected, the tree version has nodes whose data source values belong to a different reference data set than the tree.	Modify the tree version such that all nodes in the tree have data sources with reference data set matching that of the tree.
Label Enabled Node	On the Manage Tree Structures: Specify Data Sources page, if a labeling scheme is specified for the tree structure by selecting a list item from the Labeling Scheme list box, all nodes should have labels. This restriction does not apply when you select None from the Labeling Scheme list box.	The tree structure has a labeling scheme but the tree version has nodes without labels.	Assign a label to any node that does not have a label.

Date Restricted Node	On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each node in the underlying data source must have an effective date range same as the effective date range of the tree version. This restriction does not apply if the check box is not selected.	Even when the check box is selected, there are data source nodes that have a date range beyond the tree version's effective date range. For example, if the tree version is effective from Jan-01-2012 to Dec-31-2012, all nodes in the tree version must be effective from Jan-01-2012 to Dec-31-2012 at a minimum. It is acceptable for the nodes to be effective for a date range that extends partly beyond the tree version's effective date range (for example, the node data source value is effective from Dec-01-2011 to Mar-31-2013). It is not acceptable if the nodes are effective for none or only a part of the tree version's effective date range (for example, the node data source value are effective only from Jan-01-2012 to June-30-2012).	Ensure that all nodes in the tree version have effective date range for the effective date range for the tree version.
Multiple Active Tree Version	On the Manage Tree Structures: Specify Definition page, if the Allow Multiple Active Tree Versions check box is not selected for the tree structure, there should not be more than one active tree version under a tree at any time. This restriction does not apply if the check box is selected.	Even when the check box is not selected, there is more than one active tree version in the tree for the same date range.	Set no more than one tree version to Active within the same date range and set the others to inactive or draft status.
Range Based Node	On the Data Source dialog box, if the Allow Range Children check box is not selected, range-based nodes are not permitted from that data source. This restriction does not apply if the check box is selected.	Even when the check box is not selected, there are range-based nodes from a data source.	Ensure that any range nodes in your tree version are from a data source that allows range children.

Terminal Node	On the Data Source dialog box, if the Allow Use as Leaves check box is not selected, values from that data source cannot be added as leaves (terminal nodes) to the tree version. This restriction does not apply if the check box is selected.	Even when the check box is not selected, values from a data source are added as leaf nodes (terminal nodes).	Modify the tree version such that all terminal nodes are from data sources for which this check box is selected.
Usage Limit	On the Data Source dialog box, if the Use All Values option is selected to set the Usage Limit for the data source, every value in the data source must appear as a node in the tree. This restriction does not apply if None option is selected.	Even if the Use All Values option is selected, there are values in the data source that are not in the tree version.	For each data source value that is not yet available, add nodes to the tree version.

Trees and Data Sources: How They Work Together

Data sources form the foundation for tree management in Oracle Fusion Applications. Tree structures, trees, and tree versions establish direct and real-time connectivity with the data sources. Changes to the data sources immediately reflect on the **Manage Trees and Tree Versions** page and wherever the trees are being used.

Metadata

Tree structures contain the metadata of the actual data that is used in Oracle Fusion Applications. Tree structures contain the core business logic that is manifested in trees and tree versions.

Data Storage

Trees and tree versions are built upon the tree structures. They employ the business rules defined in the tree structures and allow an application to select and enable a subset of trees to fulfill a specific purpose in that application.

Access Control

Source data is mapped to tree nodes at different levels in the database. Therefore, changes you make to the tree nodes affect the source data. Access control set on trees prevents unwanted data modifications in the database. Access control can be applied to the tree nodes or anywhere in the tree hierarchy.

Adding Tree Nodes: Points to Consider

Tree nodes are points of data convergence that serve as the building blocks of a tree structure. Technically, the node may be stored either in a product-specific table or in an entity that has been established by tree management as the default

storage mechanism. However, since all data in Oracle Fusion Applications usually have a storage home, only user-created data needs to be stored in an entity.

Nodes are attached to tree versions. Whenever you create or edit a tree version, you need to specify its tree node.

Managing Tree Nodes

You can create, modify, or delete tree nodes on the **Tree Version: Specify Nodes** page. To add a tree node, ensure that the tree structure with which the tree version is associated is mapped to a valid data source. You can also duplicate a tree node if the multiple root node feature is enabled.

Node Levels

In most trees, all nodes at the same level represent the same kind of information. For example, in a tree that reflects the organizational hierarchy, all division nodes appear on one level and all department nodes on another. Similarly, in a tree that organizes a user's product catalog, the nodes representing individual products might appear on one level and the nodes representing product lines on the next higher level.

When levels are not used, the nodes in the tree have no real hierarchy or reporting structure but do form a logical summarization structure. Strictly enforced levels mean that the named levels describe each node's position in the tree. This is natural for most hierarchies. Loosely enforced levels mean that the nodes at the same visual level of indentation do not all represent the same kind of information, or nodes representing the same kind of information appear at multiple levels. With loosely enforced levels, users assign a level to each node individually. The level is not tied to a particular visual position.

Node Types

A tree node has the following node types.

- **Single:** Indicates that the node is a value by itself.
- **Range:** Indicates that the node represents a range of values and possibly could have many children. For example, a tree node representing account numbers 10000 to 99999.
- **Referenced Tree:** Indicates that the tree node is actually another version for the tree based on the same tree structure, which is not physically stored in the same tree. For example, a geographic hierarchy for the United States can be referenced in a World geographic hierarchy.

Define Profile Options

Profile Options: Explained

Profile options manage configuration data centrally and influence the behavior of applications. Profile options serve as permanent user preferences and application configuration parameters. You configure profile options with

settings for specific contexts or groups of users. Users customize how their user interfaces look and behave by changing the values of available profile options.

Profile options store the following kinds of information.

Type of Information	Profile Option Example
User preferences	Settings to provide access to social networking features
Installation information	Setting to identify the location of a portal
Configuration choices	Settings to change user interface skins and behaviors
Processing options	Settings to affect how much information to log either for an entire site or a specific user

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.

Profile Option Level	Value of the Profile Option Level	Profile Option Value	Effect
User	Manager1	UK pound sterling	Access to site and all products shows UK pounds sterling in effect

User	Manager2	US dollar	Access to site and all products shows US dollars in effect
Product	Financials for EMEA	Euro	Unless superseded by a user level value, Euros in effect for Financials for EMEA applications
Site	Site	UK pound sterling	UK pounds sterling in effect for all other users and products

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.

Hierarchy Level	Priority When Multiple Levels Set	Effect on Applications	Example
Site	Lowest	Affect all applications for a given implementation	Currency for the site is set to Euros.
Product	Supersedes Site	Affect all applications of a product family such as Financials	Currency for the Financials products set to UK pound sterling.
User	Highest, supersedes Product	Affect only the experience of the current user	Currency for the user of Financials applications set to US dollars.

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.

Tasks	Role	Considerations
Planning, creating, and editing a new profile option	Applications developer	<p>Since profile options are for permanent settings, do not use profiles options to cache temporary session attributes.</p> <p>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.</p>
Configure values in an existing profile option	Applications developer, application administrator, and implementation consultant	Manage the values for existing profile options.
Create and edit profile option categories	Applications developer, application administrator, and implementation consultant	Manage categories for organizing existing profile options.

Note

Since a profile option enables a behavior in an application user interface or across applications, a value change made by an end user is reflected in the UI page for managing profile option values.

Before Creating a Profile Option

Profile options are best defined for managing configuration data centrally and influencing the behavior of applications.

If the purpose of a profile option setting is specific to a piece of data (typically setup data), it is best implemented as an attribute of that data.

Do not use profile options for behavior that is not configurable.

Profile options exist independent of role.

Do not use profile options to implement function security. For example, an application should not check for a profile option value set to yes to provide access to a page. Do not use profile options to implement data security, such as a profile option value that must be set to a specific value to provide view access to an entity.

Do not use profile options to capture a dynamic system states, such as data stored in a temporary table. Use Global Variables for temporary states instead.

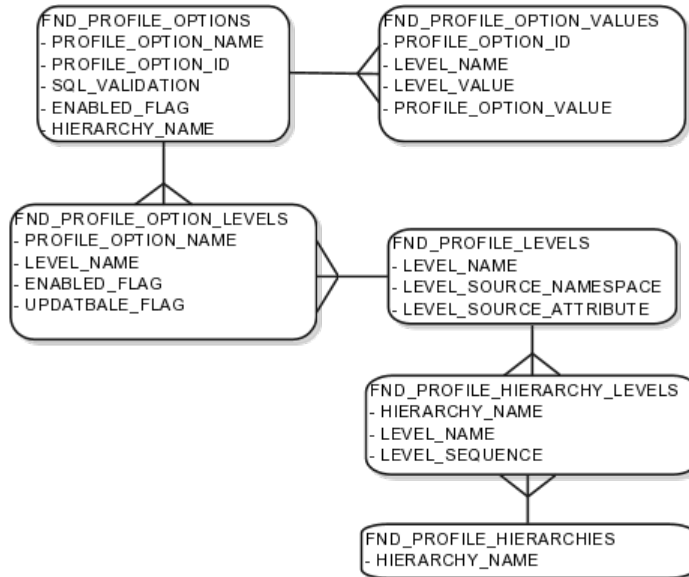
Evaluate if there is a genuine need before creating a profile option. Do not force users to make a decision about an aspect of their application use that is of no concern.

Evaluating need includes looking for duplicate or similar profile options, even in other products, before creating a new one. For example, you do not need multiple profile options to choose a preferred currency.

Profile Options Data Model

The profile option data model illustrates the relationships among profile option elements.

The figure shows the data model of profile option entities.



For more information about planning profile options, see the Oracle Fusion Applications Developer's Guide.

Managing Profile Options: Points to Consider

A profile option definition consists of a name for the profile option and valid values. It is defined within a module of the application taxonomy. Application developers manage profile options to create new profile options or modify existing profile option definitions, which includes specifying the levels at which a profile option is enabled and defining values. Implementation consultants and application administrators configure existing profile options by managing the profile option's updatable values, and creating categories that group profile options.

Configuring a Profile Option

A profile option definition includes information about the owning application and module in the application taxonomy. A start or end date, or both may limit when a profile option is active. The profile option definition may include an SQL validation statement that determines which values are valid, and the hierarchy levels at which the profile option is enabled and updatable.

To be visible to users, a profile option must be user enabled. You can also allow user updates of the profile option, which means users can make changes to the validation and the profile option level information.

Profile option levels specify at which context level profile values may be enabled or updated

Profile options should only be enabled for context levels that are appropriate for that profile option. For example, a profile option indicating a global configuration setting should not be enabled at the user level, if users cannot choose a different value for that setting.

SQL Validation

The SQL validation of the profile option definition determines what valid profile option values are available. In the absence of validation, any value is valid.

For example, SQL validation provides a means of defining a list of values for the valid values of the profile option. The SQL validation can use lookups to provide the valid values for profile options, such as the lookup codes of the YES_NO lookup type.

With a profile option called DEFAULT_LANGUAGE, you can configure the following validation.

```
SELECT DESCRIPTION Language, NLS_LANGUAGE
FROM FND_LANGUAGES_VL
WHERE INSTALLED_FLAG IN ('B','I')
ORDER BY DESCRIPTION
```

This results in the following list of values based on data in FND_LANUGUAGE_VL.

Display Value	Hidden Value
American English	US
French	F
Spanish	E

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.

Level Name	Level Value	Profile Option Value
Site	InFusion	American English

Product	Customer Center	French
Product	CRM Application Composer	American English
User	Application Administrator	American English
User	Hima	Hindi

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.

Site	Product	User	Highest Available Level	Active Profile Option Value
InFusion	CRM Application Composer	Hima	User	Hindi
Acme	Payables	Application Administrator	User	American English
InFusion	Customer Center	Guillaume	Product	French
InFusion	Payables	Implementation Consultant	Site	American English
Acme	Payables	Implementation Consultant	none	no value

Note

More than one site level value is relevant in an enterprise with multiple tenants using a single instance of Oracle Fusion Applications.

Effect of Changes to Profile Option Values

Any change you make to a user level profile option has an immediate effect on the way applications run for that session. When you sign in again, changes made to your user level profile options in a previous session are still in effect. When you change profile option value at the product level and no user level values are set, you see the update immediately, but other users may not see the changed value until signing out and back in. When you change a profile option value and the new value affects other users, the change takes effect only when users sign in the next time.

Changes to site level profile options take effect for any user session that is started after the setting has been changed. Changes to site or user level profile options do not affect any user sessions that are already in progress when the change is made.

Changes to site or user level profile options take effect for any C or PL/SQL processes, such as scheduled jobs, that are launched after the setting has been changed. Profile option changes do not affect C or PL/SQL processes that are already running.

Define Flexfields

Flexfields: Overview

Flexfields are a mechanism for configuring and extending applications to meet enterprise-specific needs by adding an attribute to a business component. Using flexfields, you can extend application objects to capture data that would not otherwise be tracked by the application, or configure intelligent key codes comprised of meaningful parts according to your business practices. Flexfields encapsulate all of the pieces of information related to a specific purpose, such as a key identifying a particular purchase, or the components of a student's contact information, or the features of a product in inventory.

A flexfield is an extensible data field consisting of subfields. Each subfield is a flexfield segment. You configure segments to capture enterprise-specific information. Each segment captures a single atomic value, has a name, and maps to a pre-reserved column in the applications database.

Flexfields allow enterprise requirements to be met without changing the data model. Different data can be captured on the same database table.

Application developers create a flexfield and register it so that it is available for configuration. Administrators and implementation consultants setup or configure segments and other properties of the available flexfields. End users see flexfield segments as 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.

Accessing Flexfields

Manage flexfields using tasks you access by starting in the Setup and Maintenance Overview page. This page is available from the Administrator menu.

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. 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.

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 expands a customizable area of an invoices page by providing subfields for collecting invoice details. You can implement these subfields, which are descriptive flexfield segments, as context sensitive so they appear only when needed. 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 additional information that is stored attributes on an entity

Segments

All flexfields consist of segments. Segments represent attributes of information. They can appear globally wherever the flexfield is implemented, or based on a structure or context.

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 predefined, 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 rather than globally available wherever the flexfield appears. (which means available to an application based on a context value rather than globally available wherever the flexfield appears. A descriptive flexfield context is a set of context-sensitive segments that store information related to the same context value. You define contexts as part of configuring a descriptive flexfield. End users see global segments, as well as any context-sensitive segments that apply to the selected context value.

Extensible flexfield segments are made available to an application based upon a category value. An extensible flexfield context serves as a container for related segments, used to organize the various segments that are applicable to a category value. You define contexts with context-sensitive segments and associate them to categories as part of configuring an extensible flexfield. End users see the segments displayed in subregions, one for each context associated to the selected category value.

In descriptive flexfields and extensible flexfields, the database columns on which context-sensitive segments are based can be reused in as many contexts as desired.

Deployment

A flexfield must be deployed to display its current definition in a runtime application user interface. For example, if the deployment status is Edited, the flexfield segments may appear in the UI based on the flexfield definition at the time of last deployment, rather than the current definition.

You can deploy a flexfield as a sandbox for testing the configuration before deploying it to the mainline for all users. In the case of extensible flexfields, you can deploy offline as a background process.

Runtime Appearance

In an application user interface, descriptive flexfield segments appear as label and field pairs or as a table of fields where the column headers correspond to the labels. The fields represent the flexfield segments and accept entered input or a selection from a list of choices that correspond to the segment's assigned value set. Extensible flexfield segments appear grouped within labeled regions, where each grouping is a context and the region labels are the context names.

Use the **Highlight Flexfields** command in the Administration menu of the Setup and Maintenance work area to identify the location of the flexfields on the runtime page. Flexfields in highlight mode display an information icon to access details about the flexfield, an edit icon to manage the flexfield, and a quick create icon to add descriptive flexfield segments.

All segments of a single flexfield are grouped together by default. The layout and positions of the flexfield segments depend on where the application developer places the flexfield on the page. Flexfields may also be presented in a separate section of the page, in a table, or on their own page or subwindow.

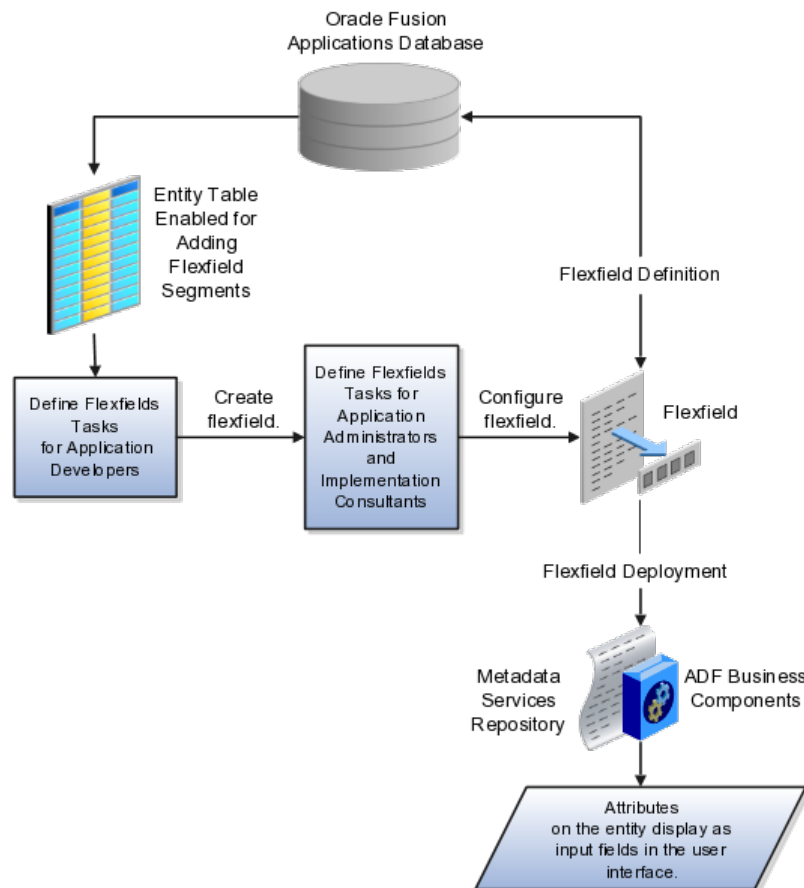
You can use Oracle Composer to edit the layout, position, or other display features of the flexfield segments.

Flexfields and Oracle Fusion Application Architecture: How They Work Together

Administrators configure flexfield segments to capture data that represents the values of attributes. Flexfield segments represent attributes of entities (business objects). Most business objects are enabled for descriptive flexfields. Some business objects are enabled for extensible flexfields.

For example, an airline manufacturer might require very specific attributes for their orders that are not provided by the out-of-the-box implementation of an order. Because a flexfield exists for the order business component, 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.

Integration

The attributes that you add by configuring flexfields are available throughout the Oracle Fusion Middleware technology stack, allowing the flexfields to be used in user interface pages, incorporated into the service-oriented architecture (SOA) infrastructure, and, in the case of descriptive flexfields, integrated with Oracle Business Intelligence.

A flexfield affects the Web Services Description Language (WSDL) schemas exposed by ADF services and used by SOA composites. The Web services that expose base entity data also expose flexfield segment data.

Attributes incorporate into SOA infrastructure (BPEL, Rules) and integrate with business intelligence (Oracle Business Intelligence, Extended Spread Sheet Database (ESSbase)).

Flexfield configurations are preserved across Oracle Fusion Applications updates.

Deployment

The metadata for the flexfield is stored in the application database as soon as you save your configuration changes. Deploying the flexfield generates the ADF business components so that the runtime user interface reflects the latest definition of the flexfield in the metadata.

Importing and Exporting

You can export and import flexfields with a deployment status of Deployed or Deployed to Sandbox across instances of Oracle Fusion Applications using the Setup and Maintenance Overview page. Ensure a flexfield is eligible for migration (by verifying that it has successfully deployed) prior to attempting the migration.

Runtime

For a flexfield to reflect the latest flexfield definition at runtime it must be deployed. The user interface accesses a business object and the deployed flexfield definition indicates which business object attributes the flexfield captures values for. If you add display customizations for a flexfield using Oracle Composer, these are customizations on the page so that the same flexfield segments can appear differently on various different pages.

Values entered for segments are validated using value sets.

Patching

Flexfield configurations are stored in an MDS repository, and are preserved during patching and upgrading.

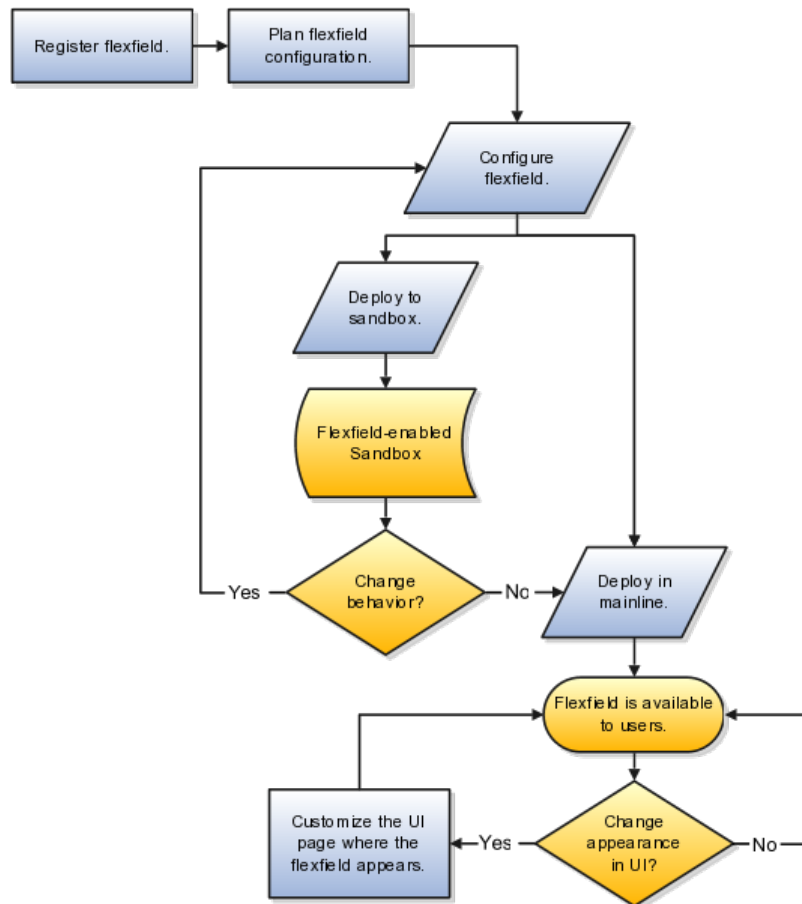
Flexfield Management

Managing Flexfields: Points to Consider

Managing flexfields involves, registering, planning, and configuring flexfields.

You plan and configure the registered flexfields provided in your applications by applications developers. How you configure flexfield segments determines how the flexfield segments appear to end users. Optionally, you can customize the UI page to change how the flexfield segments appear to end users on that page.

The figure shows the processes involved in making flexfields available to end users. The tasks in the Define Flexfields activity allow administrators to configure and deploy flexfields. If you deploy a flexfield to a sandbox and decide not to make further changes, you select the flexfield in the Manage Flexfields tasks of the Define Flexfields activity and deploy the flexfield in the mainline so it is available to users.



Registering Flexfields

Application development registers flexfields so they are available to administrators and implementation consultants for configuration.

Application development creates the capacity of database tables to support flexfields so an enterprise can capture specific information about an entity. Many flexfields are predefined in Oracle Fusion Applications.

A flexfield must be registered before it can be configured.

For more information on registering flexfields, see Oracle Fusion Applications: Developer's Guide.

Planning Flexfields

All flexfields consist of segments which represent attributes of an entity. The values an end user inputs for an attribute are stored in a column of the entity table. Application development enables columns of entity tables for use in flexfields during flexfield registration.

A flexfield must be registered before it can be configured. Before configuring new flexfield segments for your enterprise, be sure to plan their implementation carefully.

For more information on planning flexfield configuration, see Oracle Fusion Applications Extensibility Guide.

Configuring Flexfields

Administrators or implementors configure flexfields so they meet the needs of the enterprise. Some flexfields require configuration to make an application operate correctly.

You can configure flexfields using the following methods:

- Go to the manage flexfield tasks in the Setup and Maintenance work area.
- Use the Highlight Flexfields command in the Administration menu while viewing a runtime page.
 - Use the edit icon to manage a flexfield.
 - Use the quick create icon to add descriptive 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 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.

Some descriptive flexfields provide parameters that are attributes of the same or related entity objects. You use parameters to set the initial value or derivation value of an attribute from external reference data, such as a column value or a session variable, rather than from user input.

Enabling a Flexfield Segment for Business Intelligence

You can enable key flexfield segment instances and descriptive flexfield segments for business intelligence if the flexfield is registered in the database as an Oracle Business Intelligence-enabled flexfield. For more information on enabling segments for business intelligence, see points to consider when enabling key and descriptive flexfield segments for business intelligence.

Deploying Flexfields

After you configure or change a flexfield, you must deploy it to make the latest definition available to end users.

You deploy a flexfield in the mainline for general use in a test or production environment, or you can deploy a flexfield as a flexfield-enabled sandbox to confirm that it is correctly configured before deploying it to the mainline.

Deploying a flexfield results in a deployment status. Once deployed, the deployment status indicates the state of the currently configured flexfield relative to the last deployed definition

Optionally Changing How Flexfield Segments Appear in a User Interface Page

The flexfield attributes you define integrate with the user interface pages where users access the attributes' business object. Application development determines the UI pages where business objects appear and the display patterns used by default to render flexfield segments.

You can customize the appearance of the flexfield segments in the UI page using Oracle Composer once the flexfield is deployed to the mainline.

Identifying Flexfields on a Runtime Page

The **Highlight Flexfields** command in the Administration menu of the Setup and Maintenance work area identifies the location of descriptive and key flexfields on the runtime page by displaying an information icon for accessing details about each flexfield.

Even if a flexfield has not yet been deployed and does not appear on the runtime page in normal view, it appears in the Highlight Flexfield view for that page. **Highlight Flexfields** accesses the current flexfield metadata definition.

Use the highlighted flexfield's edit icon to manage the descriptive or key flexfield directly. Alternatively, note a highlighted flexfield's name to search for it in the tasks for managing flexfields.

Flexfield Segment Properties: Explained

Independent of the value set assigned to a segment, segments may have properties such as validation.

Range Validation of Segments

Range validation enables you to enforce an arithmetic inequality between two segments of a flexfield. For example, a product must be ordered before it can be shipped, so the order date must be on or before the ship date, and consequently the order date segment value must be less than or equal to the ship date segment value. You can use range validation to ensure this relationship.

The conditions for range validation are as follows.

- Segments must be configured for range validation in pairs, one with the low value and one with the high value.
- Both segments must be the same data type.
- Both segments must be part of the same structure in a key flexfield or part of the same context in a descriptive flexfield or extensible flexfield.
- The low value segment must have a lower sequence number than the high value segment.
- Non-range validated segments can exist between a range validated pair, but range validated pairs cannot overlap or be nested.

You can configure as many range validated pairs as you want within the same flexfield. Your application automatically detects and applies range validation to the segment pairs that you define, in sequence order. It must encounter a low value segment first, and the next range validated segment it encounters must be a high value segment. These two segments are assumed to be a matching pair. The low value and the high value can be equal.

Code Naming Conventions

When entering codes such as segment code, enter a code consisting of the characters A-Z, a-z, 0-9, with a non-numeric leading character.

The code is used in the flexfield's element in the XML schema for web services. You can maximize the readability of the schema by naming codes with a leading alphabetic character followed by alphanumeric characters. The use of spaces, underscores, multi-byte characters, and leading numeric characters, which are all encoded in XML schemas, make the codes in the schema element difficult to read.

This applies to context code in descriptive and extensible flexfields, and to structure instance code in key flexfields.

Flexfields and Value Sets: How They Work Together

Value sets are specific to your enterprise. When gathering information using flexfields, your enterprise's value sets validate the values your users enter based on how you defined the value set.

You can assign a value set to any number of flexfield segments in the same or different flexfields. Value set usage information indicates which flexfields use the value set.

Defining and Assigning Value Sets

As a flexfield guideline, define value sets before configuring the flexfield, because you assign value sets to each segment as you configure a flexfield.

Caution

Be sure changes to a shared value set are compatible with all flexfield segments using the value set.

Shared Value Sets

When you change a value in a shared value set, the change affects the value set for all flexfields that use that value set. The advantage of a shared values set is that a single change propagates to all usages. The drawback is that the change shared across usages may not be appropriate in every case.

Deployment

When you deploy a flexfield, the value sets assigned to the segments of the flexfield provide end users with the valid values for the attributes represented by the segments.

Defaulting and Deriving Segment Values: Explained

To populate a flexfield segment with a default value when a row is created, specify a default type of constant or parameter and a default value.

To synchronize a segment's value with another field's value whenever it changes, specify the derivation value to be the flexfield parameter from which to derive the attribute's value. Whenever the parameter value changes, the attribute's value is changed to match. If you derive an attribute from a parameter, consider making the attribute read-only, as values entered by users are lost whenever the parameter value changes.

When defaulting or deriving a default value from a parameter, only those attributes designated by development as parameters are available to be chosen.

Different combinations of making the segments read only or editable in combination with the default or derivation value or both, have different effects.

Initial runtime behavior corresponds to the row for the attribute value being created in the entity table. If the default value is read only, it cannot subsequently

be changed through the user interface. If the default value is not read only, users can modify it. However, if the segment value is a derived value, a user-modified segment value is overwritten when the derivation value changes.

Default Type	Default value specified?	Derivation value specified?	Initial runtime behavior	Runtime behavior after parameter changes
None	No	Yes	No initial segment value	Changed parameter derivation value updates segment value
Constant	Yes	No	Default segment value	N/A
Constant	Yes	Yes	Default segment value	Changed parameter derivation value updates segment value
Parameter	Yes	No	Default segment value is the parameter's default value	N/A
Parameter	Yes	Yes, and same as default value	Default segment value is parameter's default and derivation value	Changed parameter derivation value updates segment value
Parameter	Yes	Yes, and different from default value	Default segment value is parameter's default value	Changed parameter default value does not update segment value. Only the changed derivation value updates the segment value.

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 segments where the value set is assigned.

Flexfield Deployment

Flexfield Deployment: Explained

To use a flexfield at runtime, the flexfield must have been deployed at least once. Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. Flexfields are deployed for the first time during the application provisioning process.

After you configure or change a flexfield, you must deploy it to make the latest definition available to end users. You can deploy a flexfield to a sandbox for testing or to the mainline for use. In the case of extensible flexfields, you can deploy offline as a background process.

Deployment Status

Every flexfield has a deployment status.

A flexfield can have the following deployment statuses.

Deployment Status	Meaning
Edited	The flexfield metadata definition has not been deployed yet. Updates of the metadata definition are not applied in the runtime environment yet.
Patched	The flexfield metadata definition has been modified through a patch or through a data migration action, but the flexfield has not yet been deployed so the updated definition is not reflected in the runtime environment.
Deployed to Sandbox	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.
Deployed	The current metadata for the flexfield is deployed in ADF artifacts and available to end users. There have not been any changes to the flexfield since it was last deployed in the mainline.
Error	The deployment attempt in the mainline failed.

Note

Whenever a value set definition changes, the deployment status of a flexfield that uses that value set changes to edited. If the change results from a patch, the deployment status of the flexfield changes to patched.

Initial Deployment Status of Flexfields

The Oracle Fusion Applications installation loads flexfield metadata into the database. This initial load sets the flexfield status to Edited to indicate that the flexfield has not been deployed yet. The application provisioning process during installation deploys the predefined flexfields of the provisioned applications, which sets their status to Deployed if no errors are encountered.

When accessing a provisioned application, deployed flexfields are ready to use. In some cases, flexfield availability at runtime requires setup, such as defining key flexfields.

Metadata Validation

Use the Validate Metadata command to view possible metadata errors before attempting to deploy the flexfield. Metadata validation is the initial phase of 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. Errors in the metadata validation phase of deployment cause the deployment attempt to abort. Metadata validation results do not affect the deployment status of a flexfield.

Flexfield Deployment Status: How It Is Calculated

Flexfield deployment status indicates how the flexfield metadata definition in the Oracle Fusion Applications database relates to the Application Development Framework (ADF) business components generated into a Metadata Services (MDS) repository.

Settings That Affect Flexfield Deployment Status

If you have made a change to a flexfield and expect a changed deployment status, be sure you have saved your changes. No settings affect flexfield deployment status.

How Flexfield Deployment Status Is Calculated

If the flexfield definition has been edited through the Define Flexfields activity task flows, the status is Edited. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. Any change, including if a value set used in a flexfield changes, changes the deployment status to Edited. If a flexfield has never been deployed, its status is Edited.

Note

When an application is provisioned, the provisioning framework attempts to deploy all flexfields in that application.

If you deploy the flexfield to a sandbox successfully, the status is Deployed to Sandbox. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a sandbox MDS repository. Whether the sandbox is active or not does not affect the deployment status. If the flexfield was deployed to a sandbox and has not been edited or re-deployed to the mainline since then, the status remains Deployed to Sandbox independent of whether the sandbox is active, or who is viewing the status.

If you deploy the flexfield successfully, meaning to the mainline, the status is Deployed. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a mainline MDS repository. Change notifications are sent when a flexfield is deployed successfully to the mainline.

If either type of deployment fails so the current flexfield definition is not deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the Oracle Fusion application 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 are not yet in effect.

When a deployment attempt fails and you can access the Deployment Error Message for details.

Deploying a Flexfield-Enabled Sandbox: How It Works With Mainline Metadata

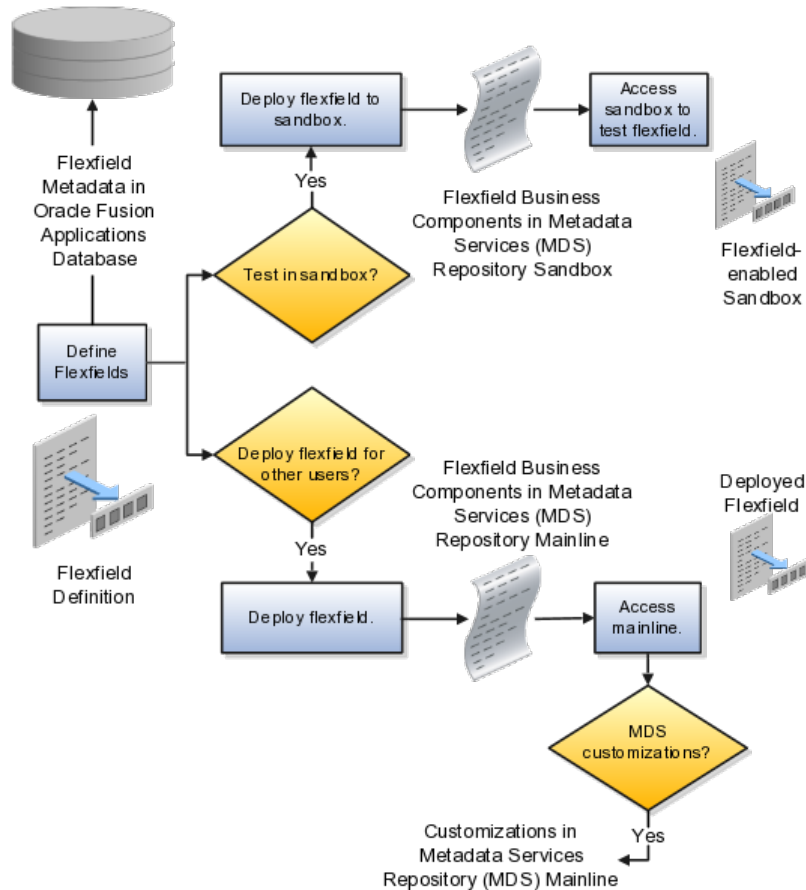
The flexfield definition in a sandbox corresponds to the flexfield metadata definition in the Oracle Fusion Applications database at the time the flexfield was deployed to the sandbox. When the flexfield is ready for end users, the flexfield must be deployed to the mainline.

A flexfield-enabled sandbox uses the following components.

- Flexfield metadata in the Oracle Fusion Applications database
- Flexfield business components in a sandbox Metadata Services (MDS) repository
- User interface customizations for the flexfield in the mainline MDS repository

The figure shows the two types of deployment available in the Manage Flexfield tasks of the Define Flexfields activity. Deploying a flexfield to a sandbox creates a sandbox MDS repository for the sole purpose of testing flexfield behavior. The

sandbox is only accessible to the administrator who activates and accesses it, not to users generally. Deploying a flexfield to the mainline applies the flexfield definition to the mainline MDS repository where it is available to end users. After deploying the flexfield to the mainline, customize the page where the flexfield segments appear. Customization of the page in the sandbox MDS repository cannot be published to the mainline MDS repository.



Sandbox Metadata Services Repository Data

Deploying the flexfield to a sandbox generates the Application Development Framework (ADF) business components of a flexfield in a sandbox MDS repository for testing in isolation.

Warning

Do not make changes to flexfield segment display features in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline.

Mainline Metadata Services Repository Data

The Oracle Fusion Applications database stores the single source of truth about a flexfield. From this the ADF business component objects that implement the flexfield in the runtime user interface are generated in the mainline MDS repository when the flexfield is deployed.

Deploying a Flexfield-Enabled Sandbox: Points to Consider

Deploying a flexfield to a sandbox creates a flexfield-enabled sandbox . Each flexfield-enabled sandbox contains only one flexfield.

You can test the runtime behavior of a flexfield in the flexfield-enabled sandbox. If changes are needed, return to the Define Flexfield tasks to change the flexfield definition.

When you deploy a flexfield to sandbox, the process reads the metadata about the segments from the database, generates flexfield Application Development Framework (ADF) business component artifacts based on that definition, and stores in the sandbox only the generated artifacts derived from the definition.

Sandbox MDS Repository Data

The sandbox data allows you to test the flexfield in isolation without first deploying it in the mainline where it could be accessed by users.

Warning

Do not make changes to flexfield segment display features in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline.

Managing a Flexfield-Enabled Sandbox

When you deploy a flexfield as a sandbox, that flexfield-enabled sandbox automatically gets activated in your user session. When you sign back in to see the changes, the sandbox is active in your session.

You can only deploy a flexfield to a sandbox using the Define Flexfields task flow pages.

You also can use the Manage Sandboxes feature in the Administration menu of the Setup and Maintenance work area to activate, access, or delete a flexfield-enabled sandbox.

Note

Whether you use the Define Flexfields or Manage Sandboxes task flows to access a flexfield-enabled sandbox, you must sign out and sign back in before you can see the changes you deployed in the runtime.

You cannot publish the flexfield from the sandbox to the mainline. You must use the Define Flexfields task flow pages to deploy the flexfield for access by users of the mainline because the flexfield configuration in the mainline is the single source of truth.

Deploying Flexfields Using the Command Line: Explained

You can use the Manage Key Flexfields, Manage Descriptive Flexfields, and Manage Extensible Flexfields tasks to deploy flexfields. You can also use WebLogic Server Tool (WLST) commands for priming the Metadata Services (MDS) repository with predefined flexfield artifacts and for deploying flexfields.

The table describes the available commands.

WebLogic Server Tool Command	Description
<code>deployFlexForApp</code>	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.
<code>deployFlex</code>	Deploy a single flexfield regardless of deployment status
<code>deployPatchedFlex</code>	Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.
<code>deleteFlexPatchingLabels</code>	Displays MDS label of flexfield changes for viewing and deleting patching labels.
<code>validateFlexDeploymentStatus</code>	Displays list including flexfields that are not deployed or failed deployment.

Executing these commands outputs a report at the command line. The report provides the following information for every flexfield that is processed.

- Application identity (APPID)
- Flexfield code
- Deployment result, such as success or error

In case of errors, the report lists the usages for which the errors were encountered. If a runtime exception occurs, the output displays the traceback information. For each WLST flexfield command, adding the `reportFormat='xml'` argument returns the report as an XML string.

Consider the following aspects of command line deployment.

- Preparing to use the WLST flexfield commands
- Using the `deployFlexForApp` command
- Using the `deployFlex` command
- Using the `deployPatchedFlex` command
- Using the `deleteFlexPatchingLabels` command
- Using the `validateFlexDeploymentStatus` command

- Exiting the WLST and checking the results

Preparing To Use the WLST Flexfield Commands

You can only execute the WLST flexfield commands on a WebLogic Administration Server for a domain that has a running instance of the Oracle Fusion Middleware Extensions for Applications (Applications Core) Setup application.

For more information on deploying the Applications Core Setup application, see the Oracle Fusion Applications Developer's Guide.

Ensure that the AppMasterDB data source is registered as a JDBC data source with the WebLogic Administration Server and points to the same database as the ApplicationDB data source.

Start the WebLogic Server Tool (WLST) tool, if it is not 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 does not deploy flexfields that have a status of Deployed unless the `force` parameter is set to `'true'` (the default setting is `'false'`).

For more information on priming the MDS partition with configured flexfield artifacts, see the Oracle Fusion Applications Developer's Guide.

From the WLST tool, execute the following commands to deploy the artifacts to the MDS partition, replacing `product_application_shortname` with the application's short name wrapped in single-quotes.

```
deployFlexForApp('product_application_shortcode', '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 do not provide a second argument.

To deploy all flexfields regardless of their deployment status, set `force` to `'true'` (the default setting is `'false'`). If you want to deploy all flexfields in a single-tenant environment, you either can set `enterprise_id` to `'None'`, or you can use the following signature:

```
deployFlexForApp(applicationShortName='product_application_shortcode', force='true')
```

Tip

The application's short name is the same as the application's module name.

For more information about working with application taxonomy, see the Oracle Fusion Applications Developer's Guide.

Using the `deployFlex` Command

From the WLST tool, execute the following command to deploy a flexfield, replacing `flex_code` with the code that identifies the flexfield, and replacing `flex_type` with the flexfield's type, which is either DFF, KFF, or EFF.

```
deployFlex('flex_code', 'flex_type')
```

The values must be wrapped in single-quotes.

Using the `deployPatchedFlex` Command

Use the `deployPatchedFlex` command for situations where the patching framework does not invoke the command, such as when an application has been patched offline.

If the installation is multi-tenant enabled, the command deploys all patched flexfields for all enterprises. This command is not intended to be invoked manually.

Check with your provisioning or patching team, or the task flows for managing flexfields, to verify that the flexfield has a Patched deployment status.

From the WLST tool, execute the following command to deploy the artifacts to the MDS partition of all flexfields that have a READY status.

```
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 ()
```

Only the flexfields that belong to the current instance of provisioned Java EE applications are checked.

Exiting the WLST and Checking the Results

To exit the tool, execute the following command.

```
disconnect ()
```

Optionally, sign into the application, access user interface pages that contain flexfields, and confirm the presence of flexfields for which configuration exists, such as value sets, segments, context, or structures.

Manage Value Sets

Value Sets: Explained

A value set is a set of valid values that you assign to a flexfield segment.

An end user enters a value into a flexfield segment while using the application. The flexfield validates the segment against the set of valid values that you configured as a value set and assigned to the segment.

For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

Caution

Be sure changes to a shared value set are compatible with all flexfields segments using the value set.

Defining value sets involves making decisions about the following.

- Validation

- Security
- Precision and scale
- Usage and deployment

Validation

The following types of validation are available for value sets.

- Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set does not present a list of valid values to users.

You can build a tree structure from the values in an independent value set whose data type is character.

Note

Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

For more information, see the Oracle Fusion Applications Extensibility Guide.

Security

Value set security only works in conjunction with usage within flexfield segments. If a value set is used standalone, meaning outside a flexfield, value set security is not applied, but Oracle Fusion data security is enforced.

You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

Value set security applies at the value set level. If a value set is secured, every usage of it in any flexfield is secured. It is not possible to disable security for individual usages of the same value set.

Value set security applies to independent, dependent, or table-validated value sets.

Value set security applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security does not determine which descriptive flexfield data is shown upon querying.

Security conditions defined on value sets 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.

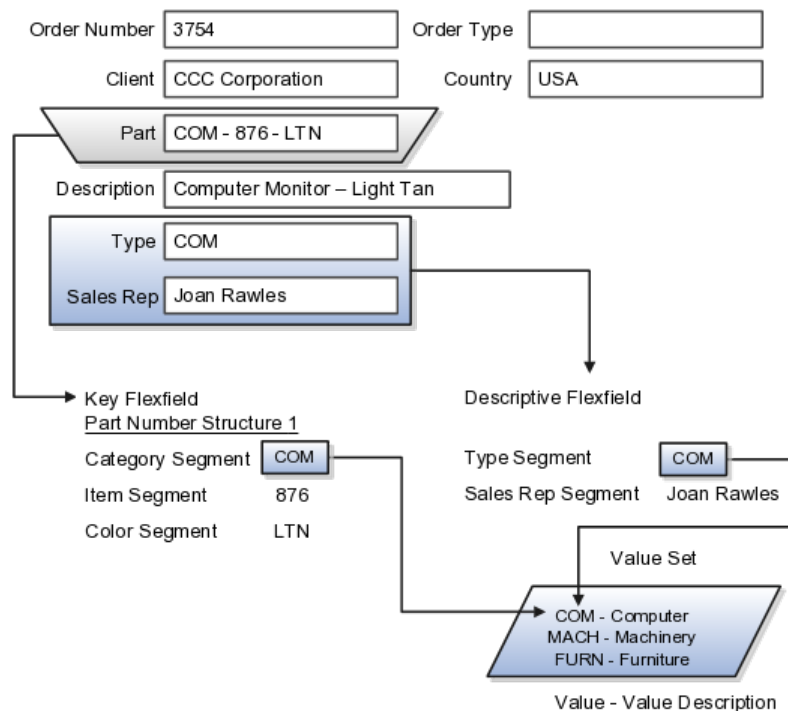
Precision and Scale

For a value set with the data type Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

Usage and Deployment

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.

The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.



For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

Defining Value Sets: Critical Choices

Validation and usage of value sets determine where and how end users access valid values for attributes represented by flexfield segments.

Tip

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.

Value Sets for Context Segments

When assigning a value set to a context segment, you can only use table-validated or independent value sets. The data type must be character and the maximum length of the values being stored must not be larger than column length of the context.

Format Only Validation

The format only validation type enables end users to enter any value, as long as it meets your specified formatting rules. That is, the value must not exceed the maximum length you define for your value set, and it must meet any format requirements for that value set.

For example, if the value set allows only numeric characters, your user could enter the value 456 (for a value set with maximum length of three or more), but could not enter the value ABC. A format only value set does not otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how many digits should follow the radix separator.

Interdependent Value Sets

You cannot specify a dependent value set for a given segment without having first defined an independent value set that you apply to another segment in the same flexfield. You use a dependent value set to limit the list of values for a given segment based on the value that the end user has chosen for a related independent segment. The available values in a dependent list and the meaning of a given value depend on which value was selected for the independently validated segment.

For example, you could define an independent value set of U.S. states with values such as CA, NY, and so on. Then you define a dependent value set of U.S. cities, with values such as San Francisco and Los Angeles that are valid for the independent value CA, and New York City and Albany that are valid for the independent value NY. In the UI, only the valid cities can be selected for a given state.

Because you define a subset value set from an existing independent value set, you must define the independent value set first. End users do not need to choose a value for another segment first to have access to the subset value set.

Table Validation

Typically, you use a table-validated set when the values you want to use are already maintained in an application table (for example, a table of vendor names). Table validation allows you to enable a segment to depend upon multiple prior segments in the same context or structure.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it does not have bind variables. If it has bind variables, the assumption is that the values are unique in the value set.

Range

In the case of format, independent, or dependent value sets, you can specify a range to further limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of segments where one segment represents the low end of the range and another segment represents the high end of the range

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100. If you use a table value set, you cannot reference flexfield segments in the WHERE clause of the value set. For example, the WHERE clause cannot reference a segment or a value set.

Security

In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

When you enable security on a table-validated value sets, the security rule that is defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set should not assume anything about the bind variables; it must assume the whole list of values is available and write the rule, for example, to allow x, or to allow y and z. By default in data security all values are denied, and show only rows to which access has been provided.

Maintenance

There is no need to define or maintain values for a table-validated or subset value set, as the values are managed as part of the referenced table or independent value set, respectively.

If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select **Translatable**. This does not require you to provide translated values now, but you cannot change this option if you decide to provide them later.

For more information about defining value sets, see the Oracle Fusion Applications Extensibility Guide.

Manage Descriptive Flexfields

Descriptive Flexfields: Explained

Descriptive flexfields provide a way to add custom attributes to entities, and define validation and display properties for them. These attributes are generally standalone. They don't necessarily have anything to do with each other and are not treated together as a combination.

All Oracle Fusion Applications business entities that you can access are enabled for descriptive flexfields. Descriptive flexfields are optional. You can choose to configure and expose segments for the descriptive flexfield defined and registered in your database, or not. 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. The segments of a descriptive flexfield are made 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.

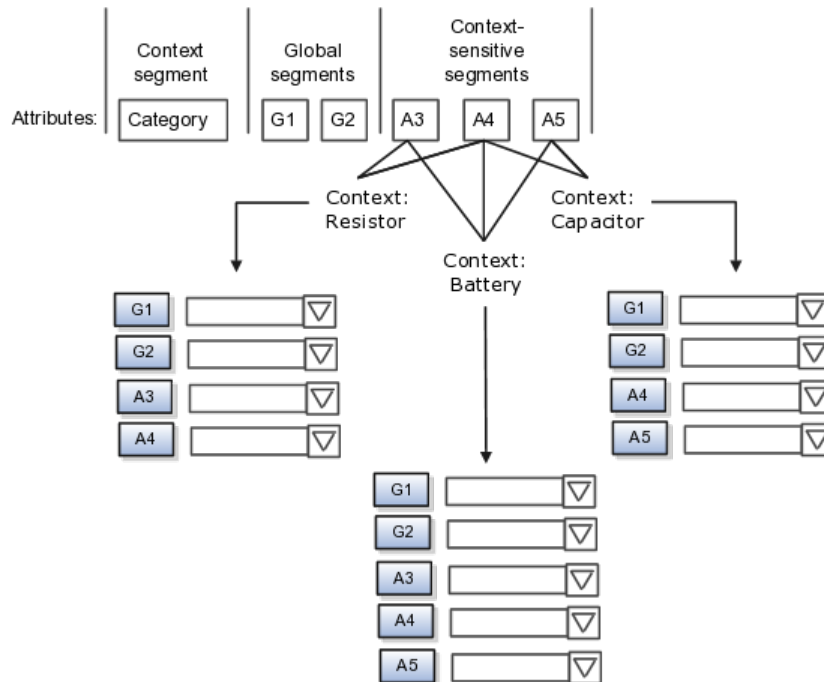
The same underlying column can be used by different segments in different contexts. For example, you can define a Dimensions context that uses the ATTRIBUTE1 column for height, the ATTRIBUTE2 column for width, and the ATTRIBUTE3 column for depth. You can also define a Measurements context that uses the 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.

Segment Type	Runtime Behavior
Global segment	Always available
Context segment	Determines which context-sensitive segments are displayed
Context-sensitive segment	Displayed depending on the value of the context segment

In the figure, a descriptive flexfield has one context segment called Category for which there are three values: Resistor, Battery, and Capacitor. In addition, the descriptive flexfield consists of two global segments that appear in each of the contexts, and three context-sensitive segments that only appear in the context in which they are configured.



Application development determines the number of segments available for configuring. During implementation you determine which attributes to add using the available segments, and the context values and the combination of attributes in each context. A segment can be used for different attributes, such as Height in Context1 and Color in Context2. Each segment of a descriptive flexfield that you make available to end users is exposed in the user interface as an individual field.

Value Sets

For each global and context-sensitive segment, you configure the values allowed for the segment and how the values that end users enter are validated, including interdependent validation among the segments.

Managing Descriptive Flexfields: Points to Consider

Configuring descriptive flexfields involves managing the available flexfields registered with your Oracle Fusion Applications database and configuring their flexfield-level properties, defining and managing descriptive flexfield contexts, and configuring global and context-sensitive segments.

Every descriptive flexfield is registered to include a context segment, which you may choose to use or not.

Segments

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. If you do not specify a value set for a context segment, the valid values for that context segment are derived from the context codes. The definition of each context segment specifies the set of context-sensitive segments that can be presented when that context code is selected by the end user.

For reasons of data integrity, you cannot delete an existing context. Instead, you can disable the associated context value in its own value set by setting its end date to a date in the past.

You can configure the individual global segments and context-sensitive segments in a descriptive flexfield. These segment types are differentiated by their usage, but they are configured on application pages that use most of the same properties.

Usages

Descriptive flexfield usages allow for the same definition to be applied to multiple entities. Descriptive flexfield tables define the placeholder entity where the flexfield segment values are stored once you have configured the descriptive flexfield.

Parameters

Parameters are public arguments to a descriptive flexfield. Parameters provide outside values in descriptive flexfield validation. Parameters can be referenced by the logic that derives the default segment value and in table-validated VALUE set WHERE clauses.

Delimiters

A segment delimiter or separator visually separates segment values when the flexfield is displayed as a string of concatenated segments.

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 context-sensitive segments to prevent duplication and complexity in the flattened components
- 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, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly-generated attributes in business intelligence applications. For additional information about import, refer to the Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

Flattening

When you deploy a business intelligence-enabled descriptive flexfield, the deployment process generates an additional set of flattened business components for use in Oracle BI. The flattened business components include attributes for business intelligence-enabled segments only.

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, if you assigned a label to the segment, the flattened components include an additional single attribute representing segments with that label. If you did not assign a label, the flattened components include a discrete attribute for each BI-enabled context-sensitive segment in each context.

Equalization

You can prevent duplication and the extra workload and complexity that result from the flattening process by assigning a label to any set of context-sensitive segments that serve the same purpose in different contexts. BI-enabled context-sensitive segments that have labels are equalized so that the flattened business components include one attribute for each label, consolidated across contexts. Non-labeled context-sensitive segments are not 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.

Mapping to Logical Objects in Business Intelligence

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 does not 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.

Manage Extensible Flexfields

Extensible Flexfields: Explained

Extensible flexfields are like descriptive flexfields, with some additional features.

- You can add as many context-sensitive segments to the flexfield as you need. You are not dependent on the number of segments predefined and registered for the flexfield.
- You can configure a one-to-many relationship between the entity and its extended attribute rows.
 - A row of data can have multiple contexts associated with it.
 - A row of data can have multiple occurrences of the same context.
- You can configure contexts in groups so the attributes in the context always appear together in the user interface.
- You can use existing hierarchical categories so that entities inherit the contexts that are configured for their parents. Contexts are reusable throughout categories.
- You can specify view and edit privileges for the extensible flexfield segments to control who sees the attributes and who can change the attribute's values.

When you configure a context for multiple rows per entity, the segments are displayed as a table.

Unlike descriptive flexfields, the extension columns corresponding to extensible flexfields segments are part of extension tables, separate from the base application table. Unlike descriptive flexfield contexts, the set of attributes in an extensible flexfield context remains constant and does not differ by context value.

An extensible flexfield describes an application entity, with the runtime ability to expand the database that implementation consultants can use to define the data structure that appears in the application.

Extensible flexfields support one-to-many relationships between the entity and the extended attribute rows.

For lists of extensible flexfields, see assets with the Flexfield: Extensible type in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>)

Usages

As with descriptive flexfields, you can define multiple usages for an extensible flexfield, which enables several application tables to share the same flexfield.

For example, a flexfield for shipping options can be used by both a Supplier table and a Buyer table. In addition, you can associate a context with one, some, or all of the flexfield's usages. Thus, with the shipping information example, you can associate a warehouse context with the Supplier usage, a delivery location context with the Buyer usage, and a ship-via context with all usages.

Categories

Extensible flexfields support multiple contexts, and the contexts can be grouped into categories. Every extensible flexfield has at least one category, or root category, and some extensible flexfields support a hierarchy of categories, where a given category can inherit contexts from its parent categories.

You can define categories for extensible flexfields, and you can associate any combination of contexts with a given category.

For example, the Electronics and Computers category hierarchy might include a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on. The Home Entertainment product might have contexts that specify voltage, dimensions, inputs and outputs. Contexts are reusable within a given extensible flexfield. For example, the dimensions context could be assigned to any category that needs to include dimensional information.

Pages

Extensible flexfields allow you to combine contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

Each application page corresponds to one extensible flexfield category, with a separate region of the page for each associated context.

Managing Extensible Flexfields: Points to Consider

Configuring extensible flexfields involves managing the available flexfields registered with your application database and configuring their flexfield-level properties, defining contexts, categories, and pages, and configuring the segments for each extensible flexfield.

Contexts

A context can be defined as single row or multi row. Single row contexts are the same as descriptive flexfields contexts. A single row context has only one set of context-sensitive segments. A multi-row context enables you to associate multiple sets of values with the same object instance.

For example, for a BOOK table, you could create a multi-row context named chapters that contains a chapter segment and a number of pages segment. Multiple chapters can then be associated with each book in the BOOK table.

Set the context to translatable so free-form text entered by end users is stored in the language of the user's locale, and different translations of that text can be stored in other languages. Segments in the translated contexts should utilize format-only value sets for storing free-form, user-entered text.

Categories

A category is a grouping of related data items that can be considered to belong together. You can associate any combination of contexts with a given category. Extensible flexfields with more than 30 categories must be deployed using the Deploy Offline command.

A category hierarchy logically organizes a set of categories.

For example, the Electronics and Computers category hierarchy might include a Computer category and a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on.

A category can be a child or sibling of an existing category. The hierarchy can be as simple or as complex as desired, with any combination of zero or more sibling categories and zero or more child categories. If no category is defined, the data items are grouped under a single predefined default category.

Each category has associated contexts that store relevant information about a data item in that category. For example, a Home Entertainment product has contexts that specify Voltage, Dimensions, Inputs and Outputs. Contexts are reusable within a given extensible flexfield; the Dimensions context could be assigned to any category that needs to include dimensional information.

If a hierarchy includes child categories, each child category inherits the contexts from its parent category; for example, the Home Entertainment category inherits Voltage and Dimensions from the Electronics and Computers category.

Each extensible flexfield is associated with a particular category hierarchy. You can think of category hierarchies as the defining framework for extensible flexfields and their contexts. A category hierarchy specifies which contexts are valid for each category.

An extensible flexfield can include multiple contexts which you define to support a given category. These contexts can be suitable for a variety of different purposes, but within a particular category, some contexts might be considered to be related to, or dependent on, each other. You can combine these contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

For example, the Home Entertainment category might have an Electrical Specifications page that contains the Voltage, Inputs and Outputs contexts, and a Physical Specifications page that contains the Dimensions and Form Factor contexts.

Indexed Segments

You can designate an extensible flexfield segment as indexed so that it is one of the selectively required attributes an end user can use in an attribute search. If

you indicate in the Manage Extensible Flexfield UI page that a segment should be indexed, the column representing the segment must be added to the database index. This is commonly done by a database administrator (DBA).

When an extensible flexfield with indexed segments is deployed, search task flows are generated along with the other flexfield artifacts and specify the indexed attributes as selectively required. In the deployed extensible flexfield's search task flow, an end user must specify at least one of the indexed attributes in the search criteria. This prevents non-selective searches which could cause performance issues.

For example, if you index the memory and processor attributes and ensure that the corresponding columns in the database are indexed, an end user can search an item catalog for computers by entering processor or memory or both as a search criterion. No search is performed if an end user enters an attribute as search criteria that is not indexed.

Offline Deployment

You can deploy extensible flexfields offline as a background process using the Deploy Offline command in the Manage Extensible Flexfields task, and continue working in the session without having to wait for the deployment to complete. Deploy Offline enables adding one after another extensible flexfield to your deployment queue. For extensible flexfields with more than 30 categories you must use Deploy Offline.

You can remove an extensible flexfield from the deployment queue with the Cancel Offline command.

When an extensible flexfield is deployed offline, its offline status indicates that the flexfield is in an offline deployment process. The Offline Status column refreshes when you perform a new search in the Manage Extensible Flexfields task. A flexfield's offline status is cleared and its deployment status updated when the offline deployment process has completed.

Manage Key Flexfields

Key Flexfields: Explained

Key flexfields provide a means to capture a key such as a part number, a job code, or an account code. A key flexfield consists of one or more segments, where each segment can have a meaning.

For example, a part number 10-PEN-BLA-450 might correspond to a black pen from vendor #450 sold by division #10 (office supplies). Behind the scenes, the application uses a unique number, 13452, for this part, but the end user always see the 10-PEN-BLA-450 part number.

The following aspects are important to understanding key flexfields.

- Architecture
- Segments and segment labels
- Structures
- Segment and structure instances

- Combinations
- Dynamic combination creation
- Security

Key flexfields are not optional. You must configure key flexfields to ensure that your applications operate correctly. You configure and maintain key flexfield definitions with the Manage Key Flexfields task.

For lists of key flexfields, see assets with the Flexfield: Key type in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

Architecture

When you configure a key flexfield, you define metadata about the key flexfield such as how many segments are in a structure, how many structures the flexfield uses, what value sets each segment uses, and so on. This is flexfield metadata stored in flexfield metadata tables.

Based on the flexfield metadata, actual part numbers are captured at runtime as a combination of segment values and stored in a combinations table. A combinations table contains all the segment columns for a flexfield, plus a unique ID column and a structure instance number column that differentiates multiple arrangements of the segment columns.

For example, a part number that can be comprised of multiple segments can be represented by a key flexfield. A part number key flexfield has a corresponding combinations table, where the flexfield stores a list of the complete codes, with one column for each segment of the code, together with the corresponding unique ID and structure instance number for the code. When users define a new part number or maintain existing part numbers in the parts catalog, they directly maintain rows in the combination table.

The foreign key table contains a different business entity than the combinations table. For example, the business entity in the foreign key table is order lines or invoice lines that contain foreign key references to parts for ordering and so on. Any number of foreign key tables can reference a particular entity represented by a key flexfield.

Segments and Segment Labels

A key flexfield consists of segments. Segments consist of a prompt, a short prompt, display width, a number that determines where in the sequence of a key flexfield structure the segment exists, the range type and the column name of the attribute being captured by the segment, a default value set and a label for the segment. A segment label identifies a particular segment of a key flexfield. Segment labels are defined and made available by applications development.

Applications identify a particular segment for some purpose such as security or computations. Segment name or segment order cannot reliably identify a segment because key flexfield segments can be configured to appear in any order with any prompts. A segment label functions as a tag for a segment.

For example, Oracle Fusion General Ledger needs to identify which segment in the Accounting Flexfield contains balancing information and which segment contains natural account information. General Ledger uses a segment label to

determine which segment you are using for natural account information. When you define your Accounting Flexfield, you must specify which segment label apply to which segments.

Some labels must be unique, and cannot be applied to more than one segment in each structure. Other labels are required, and must be applied to at least one segment in each structure.

A segment label orients an end user's search of segments, such as the Cost Center label for all segments across key flexfields that capture a value for cost center.

Structures

A key flexfield structure definition includes the number of segments and their order.

In some applications, different users need to see different segment structures for the same flexfield. A key flexfield can have multiple structures if registered to support more than one structure.

The flexfield can display different fields for different end users based on a data condition in your application data, such as the value of another field entered by the end user or the user's role. For example, the correctly formatted local postal address for customer service inquiries differs based on locale. A postal address key flexfield could display different segments and prompts for different end users based on a location condition in your application data, such as the user's role or a value entered by the user.

Each structure can have one or more segments. Thus a segment is a child of a structure. If you want to store a particular segment, such as Cost Center, in two different structures, you must define the segment separately in each structures.

Each structure may have one or more structure instances. Each instance of a structure shares the same number and order of segments, but differs in the allowable values or value sets that validate the segments.

Structure and Segment Instances

You can define multiple configurations of a key flexfield structure. These structure instances have the same segment structure, in the same sequence order. They differ primarily in how each segment is validated. You define a structure instance for each key flexfield and each key flexfield structure instance.

The segments in a key flexfield structure instance are segment instances. A segment instance is a segment with a specific value set assigned to it.

If a key flexfield has been registered with a tree structure, you can specify a tree code for a segment instance, where the tree code defines a hierarchical relationship between the segment values.

Combinations

A combination is a complete code, or combination of segment values that makes up the code, that uniquely identifies an object.

For example, each part number is a single combination, such as PAD-YEL-11x14 or 01-COM-876-7BG-LTN. In these combinations, the hyphen is the segment separator. If you had ten parts you would define ten combinations. A valid

combination is simply an existing or new combination that can currently be used because it is not out of date or disabled, and does not violate cross-validation or security rules. A combination has different segments depending on the flexfield structure being used for that combination. Any combination is associated with only one particular flexfield structure.

Many Oracle Fusion Applications products refer to a key flexfield combination by using the name of the entity or the key flexfield itself. For example, Oracle Fusion Assets uses the asset key flexfield and refers to one of its combinations as an asset key or asset key flexfield. In another example, other Oracle Fusion Applications products including Oracle Fusion General Ledger (GL) refer to combinations of the accounting flexfield as account or GL account.

Each key flexfield has one corresponding table, known as the combinations table, where the flexfield stores a list of the complete codes, with one column for each segment of the code, together with the corresponding unique ID number (a code combination ID number or CCID) for that code. Then, other tables in the application have a column that stores just the unique ID for the code. For example, you may have a part number code, such as PAD-YEL-11x14. The Parts combinations table stores that code along with its ID, 57494. If your application allows you to take orders for parts, you might then have an Orders table that stores orders for parts. That Orders table would contain a single column that contains the part ID, 57494, instead of several columns for the complete code PAD-YEL-11x14.

Typically one combinations page maintains the key flexfield, where the key flexfield is the representation of an entity in your application. The combinations page is where you maintain individual combinations, such as part numbers.

Dynamic Combination Creation

Dynamic combination creation is the insertion of a new valid combination into a combinations table from a page other than the combinations page.

Dynamic combination creation may be enabled at the following levels.

Level Of Dynamic Combination Creation	Controlled By:
Flexfield	Application development
Each usage or reference to the key flexfield	Application development
Structure instance	Administrators and implementation consultants
Other	Administrators and implementation consultants

If your key flexfield or certain usages or references of the key flexfield do not permit dynamic combination creation, you may control whether dynamic combination creation is enabled for each structure instance. If enabled, a user can enter a new combination of segment values using the flexfield window from a foreign key page. For example, when entering a transaction, a GL user can enter a new expense account code combination for an account that does not yet exist. Your application creates the new account by inserting the new combination into the combinations table behind the scenes. Assuming that the new combination satisfies any existing cross-validation rules, the flexfield inserts the new combination into the combinations table, even though the combinations table is not the underlying table for the foreign key page.

Managing Key Flexfields: Points to Consider

Consider the plans for a key flexfield, security, and resulting runtime pages when configuring key flexfields.

Planning

Plan structures carefully and allow for future needs.

Caution

Do not change the number, order, and maximum length of segments once you have acquired flexfield data.

Structure Delimiters

A delimiter separates the segments when they appear to end users. The delimiter value of a structure specifies the character used to visually separate segment values when the key flexfield is displayed as a string of concatenated segments in the UI.

Tip

Choose the delimiter value of your key flexfield carefully so that it does not conflict with the flexfield data. For example, if your data frequently contains periods, such as in monetary or numeric values, do not use a period as your segment separator. Any character you expect to appear frequently in your segment values or descriptions is not a good choice for the delimiter.

If you change the configuration of a key flexfield, such as the delimiter, the change affects the previously stored key flexfields with that structure.

Security

Oracle Fusion data security enforces value set security.

Within key flexfields, value set security applies to the selection of the individual segment values in the segment list of values. When selecting a key flexfield segment value from the combination table, data security allows display of only the combinations whose segment values you have access to. Applications development controls whether or not value set security rules propagate to the foreign key table. By default they do.

Runtime Pages

Application development determines the user interface (UI) pages used to render flexfields. The types of key flexfield UI pages are as follows.

- Combinations pages where underlying entity objects use the combinations table itself
- Foreign key pages where the underlying entity objects contain a foreign key reference to the combinations table
- Partial usage page where some or all of the key flexfield's segment columns are in a product table

The same key flexfield can be used in different ways on different pages.

A page with a foreign key reference has a base table or view that contains a foreign key reference to a combinations table with the actual flexfield segment columns. This allows manipulating rows containing code combination IDs (CCID).

A page with partial usage of a key flexfield presents segments that are defined on a product's transactional table in addition to being defined on a combinations table. In the case of a partial usage page, it is possible that only part of the configuration is visible. This allows the key flexfield to behave more like a descriptive flexfield.

A code combination maintenance page or combinations page presents the combinations table. This allows directly creating and maintaining code combinations. The combinations table contains all key flexfield segment columns and a unique ID column.

A typical application has one and only one combinations page. An application might not have a combinations page if it does not support maintenance by administrators.

A page containing a search region enables end users to select which attributes of the key flexfield view object to use as criteria to search for flexfield metadata.

For example, you can configure seven segments for the Account key flexfield. In a foreign key reference page, end users see the typical key flexfield picker with all seven segments where they can search for combinations. In a partial usage page using the same key flexfield, end users potentially could see only a single segment such as the Cost Center labeled segment, or they might see multiple segments but displayed as individual segments rather than as a picker for choosing combinations.

For more information on key flexfield pages, see the Oracle Fusion Applications Developer's Guide.

Key Flexfield Structures: Explained

A key flexfield structure arranges the segments of a key so you can reuse a single key flexfield in multiple combinations of the same or a subset of segments. Multiple instances of a single structure can accommodate differences in the value sets assigned to the structure's segments.

The structure determines the following aspects of a key flexfield.

- The segments to include
- The order of the segments
- Segment labels on the included segments
- Properties for each segment applied to the instances of the segments in an instance of the structure

Managing Key Flexfield Structures

All the segments defined for a key flexfield are available to be included in a key flexfield structure.

You can define as many segments as there are defined segment columns in your key flexfield combinations table.

Restriction

Be sure to add segments in the order that your key requires. Once deployed, the order cannot be changed.

Enable segments to indicate that they are in use. A flexfield does not display disabled segments in runtime.

Tip

To protect the integrity of your data, disable a segment if you have already used it to enter data.

Key Flexfield Structure Instances and Segment Instances: Explained

A key flexfield structure can have one or more alternate structure instances.

The instances of a key flexfield structure share the following aspects of the structure.

- The same set of segments
- The same arrangement of segments
- The same properties at the segment and structure levels

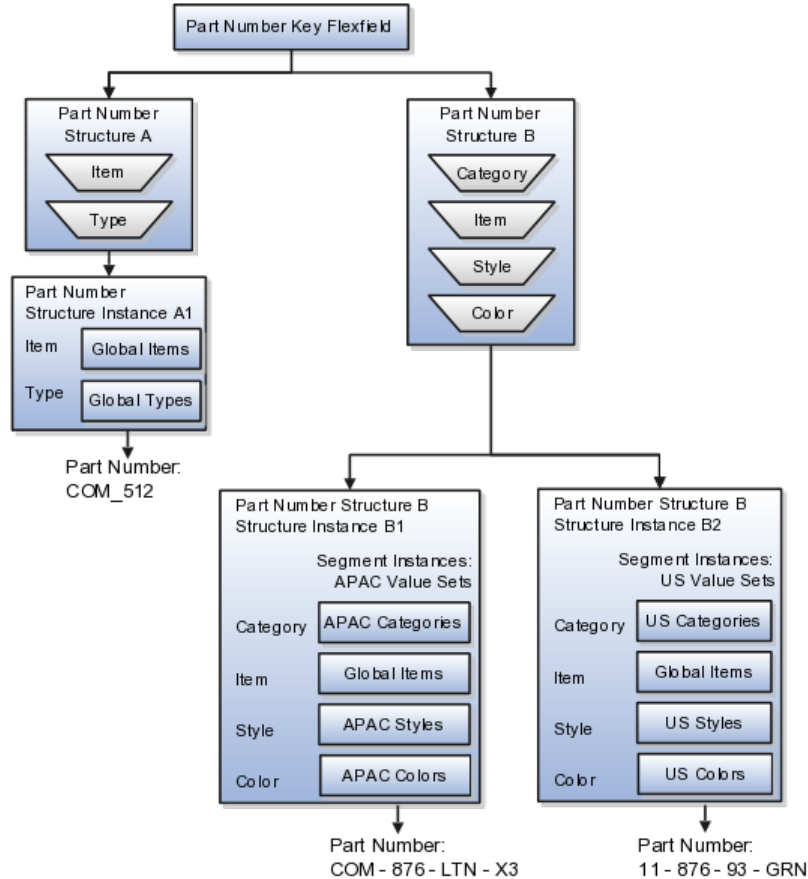
Differences among structure instances at the structure level include whether dynamic combination creation is allowed.

Differences among segment instances at the structure instance level include the following.

- Value set
- Default type and default value
- Tree code
- Whether the segment is any of the following
 - Required
 - Displayed
 - Enabled for business intelligence
 - Optional or required as a query criterion

For example, you could use one group of value sets for the US and another for France.

The figure shows two structures instances for a part number structure. The structures differ in the number of segments and the segment separators used. The structure instances of a structure share all properties that are defined for the structure, but can vary in the properties defined at the structure instance or segment instance level, such as the value set assigned to the segment instances.



Query Required Segment Instances

You can designate a key flexfield segment instance as query required so that it is one of the selectively required attributes an end user can use in a key flexfield combination search. If you indicate in the Manage Key Flexfields UI page that a segment instance should be indexed, the column representing the segment must be added to the database index. This is commonly done by a database administrator (DBA).

Following deployment, the combination picker of the key flexfield displays the query required attributes as selectively required. An end user must specify at least one of the query required attributes in the search criteria. This prevents non-selective searches that could cause performance issues.

For example, if you mark the cost center and account attributes as query required and ensure that the corresponding columns in the database are indexed, an end user can search for combinations by entering cost center or account or both as a search criterion. No search is performed if an end user does not enter at least one query required attribute as search criteria.

Tip

Index the Structure Instance Number column on your combinations table to improve performance.

Dynamic Combinations

If a key flexfield supports dynamic combination creation, you can choose to enable this feature by selecting **Dynamic Combination Creation Allowed**. This will allow end users to enter values at runtime that produce new code combinations for the flexfield. If not enabled, new valid combinations can only be entered using the combinations table for the flexfield.

Trees

If a tree code has been defined for the value set assigned to the segment instance, and you assign the tree code to the segment instance, tree hierarchy search operations are available on the segment values.

For a segment instance to be based on a tree, the following must be true.

- Application development registered the key flexfield with a tree structure.
- A tree code for that tree structure exists.
- The tree code that includes tree versions containing the values of the value set assigned to the segment instance.
- You assign the desired tree code directly to the segment instance.

Provided these conditions are satisfied, different segment instances that use the same value set can be assigned the same or different tree codes, meaning they use a different hierarchy definition over the same values.

Enabling Key Flexfield Segments for Business Intelligence: Points to Consider

A key flexfield that is registered in the database as enabled for Oracle Business Intelligence (BI) includes a BI Enabled setting for each of its segment instances. When a segment instance is BI-enabled, it is available for use in Oracle Business Intelligence.

The following aspects are important in understanding BI-enabled key flexfield segments.

- Flattening business components to use BI-enabled segments in Oracle BI.
- Equalizing segments to prevent duplication and complexity in the flattened component.
- Mapping attributes of flattened business components to logical objects in Oracle BI.
- Managing the labels that map segments to logical objects in Oracle BI.

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly generated attributes in business intelligence applications. For additional information about import, refer to the Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition (Oracle Fusion Applications Edition).

Flattening

When you deploy a business intelligence-enabled key flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you did not assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

Equalization

You can prevent duplication and the extra workload and complexity that result from the flattening process by assigning a label to any set of segments that serve the same purpose in different structures. BI-enabled segments that have labels are equalized so that the flattened business components include one attribute for each label, consolidated across structures. Non-labeled segments are not equalized across structures, so the flattened components include a separate attribute for each segment for each structure.

Note

It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Mapping to Logical Objects in Business Intelligence

Assign a label to a segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

Note

Assigning a label to a segment serves to equalize the attribute across structures, as well as map the equalized attribute to business intelligence.

Managing Labels

You may assign a predefined label (if available) to segments or create new labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across structures.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label does not exist in business intelligence, you must manually map the segment to the desired logical object when importing into Oracle Business Intelligence.

In addition, segments without labels cannot be equalized across structures. The flattened components include a separate attribute for each non-labeled segment in each structure.

Note

Segment labels serve other functions as well, as presented in Key Flexfields: Explained.

Key Flexfields: Example

A key flexfield can capture expense account information.

Scenario

When entering details for each expense, the user specifies an account to which the expense is charged.

Entering Expense Accounts

A user interface for entering expenses gives the user the option of selecting an expense account that identifies the cost center and other details needed for processing the expense.

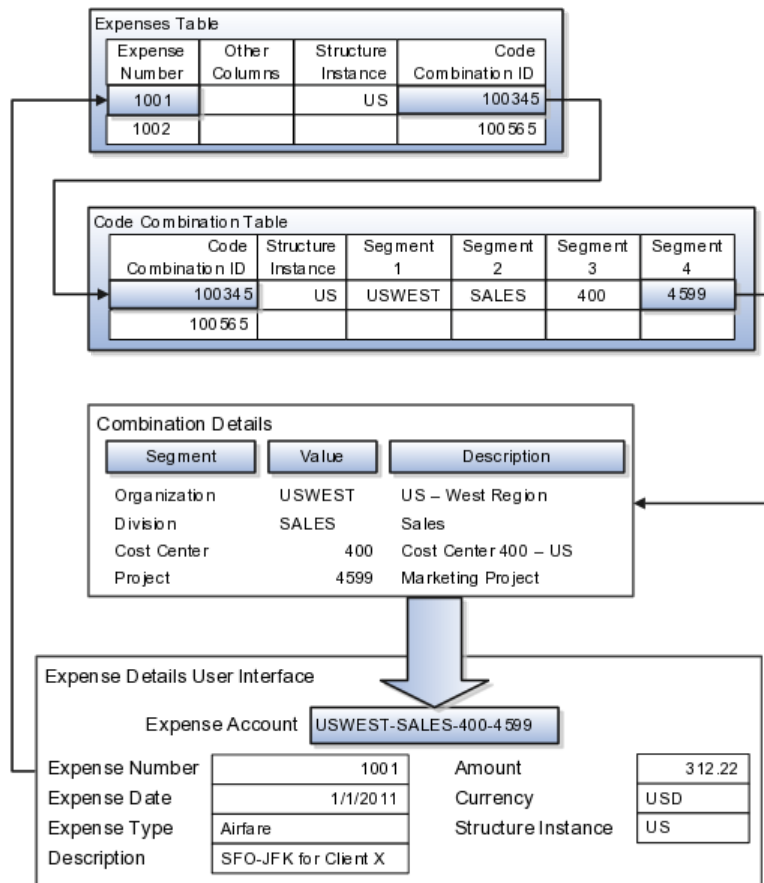
Analysis

The expense account field is a foreign key reference to a code combination (EXPENSE_LINES.EXPENSE_ACCOUNT = ACCOUNTS.CCID).

Code Combination Table for Entering Accounts and Employees

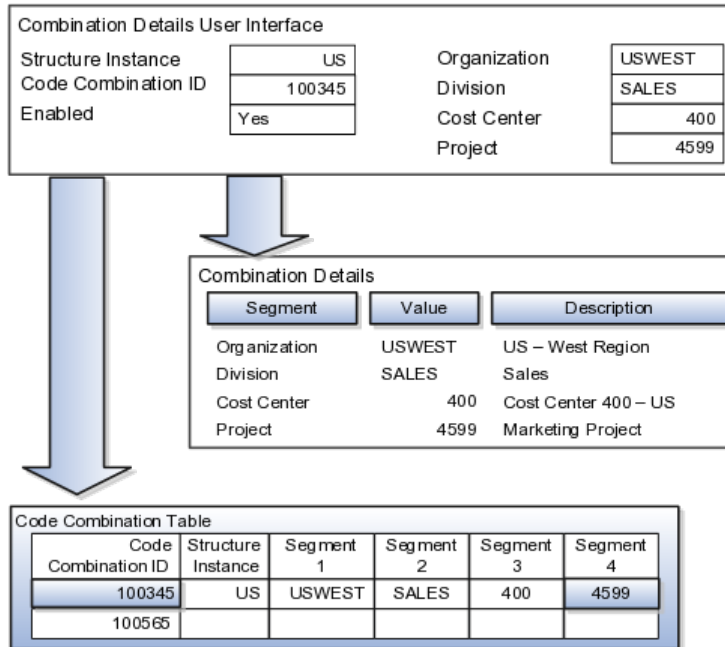
The code combination table supports entering account information, such as for expense accounts.

The figure shows the origin in the code combination table of the account specified by the user. The code combination ID record stores the information of the key flexfield segments used to assemble the expense account based on the key flexfield configuration.



The combinations page, which is the maintenance page for the key flexfield, is for managing rows in the combination table. In this example, managing the combinations means adding or editing account numbers that adhere to the key flexfield metadata rules.

The figure shows the code combination details for the example expense account reflected in the flexfield configuration and the code combination table.

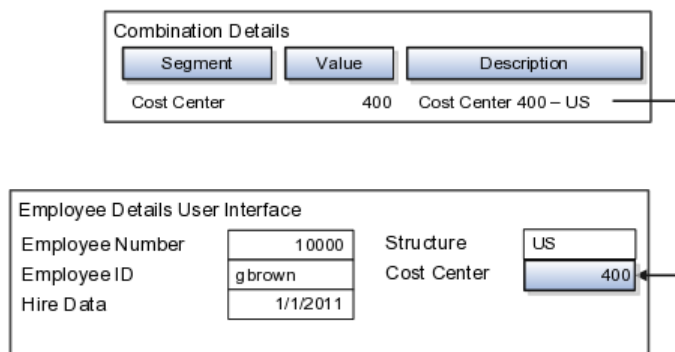


If dynamic combination creation is not enabled, then when entering an expense line, the user can only select an account that already exists in the ACCOUNTS (combinations) table. If they require an account that does not exist, they must consult with the appropriate application administrator who can add the account to the combinations table.

If dynamic combination creation is enabled, then when entering an expense line, the user can either select a pre-existing account, or type in a new account that created dynamically on the fly in the ACCOUNTS (combinations) table. Once the new combination is created, the same user can refer to it on the expense line.

When managing employee information, the user specifies the cost center that the employee belongs to. The cost center field corresponds to a single, labeled segment of the Account Key Flexfield and has metadata defined such as the allowable value set for that segment.

In this figure, instead of specifying a cost center ID reference to an account, only the Cost Center segment is used and the value is stored directly on the employee table.



FAQs for Define Flexfields

Why did my flexfield changes not appear in the runtime UI?

The ADF business components or artifacts of a flexfield, which are generated into an metadata services (MDS) repository when the flexfield is deployed, are cached within a user session. You must sign out and back in again to view flexfield definition changes reflected in the runtime application user interface page.

A flexfield's status relative to its deployment determines whether the flexfield segments as currently defined in the metadata are available to end users. The flexfield segments seen by end users in the runtime correspond to the flexfield definition that was last deployed successfully.

What happens if a value set is security enabled?

Value set security is a feature that enables you to secure access to value set values based on the end user's role in the system.

As an example, suppose you have a value set of US state names. When this value set is used to validate a flexfield segment, and users can select a value for the segment, you can use value set security to restrict them to selecting only a certain state or subset of states based on their assigned roles in the system.

For example, Western-region employees may choose only California, Nevada, Oregon, and so on as valid values. They cannot select non-Western-region states. Eastern-region employees may choose only New York, New Jersey, Virginia, and so on as valid values, but cannot select non-Eastern-region states. Value set security is implemented using Oracle Fusion Applications data security.

How can I set a default value for a flexfield segment?

When you define or edit a flexfield segment, you specify a default value from the values provided by the value set assigned to that segment.

You can set the default value to be a parameter, which means the entity object attribute to which the parameter you choose is mapped will provide the initial default value for the segment.

You can set to be a constant, if appropriate to the data type of the value set assigned to the segment.

In addition to an initial default value, you can set a derivation value for updating the attribute's value every time the parameter value changes. The parameter you choose identifies the entity object source attribute. Any changes in the value of the source attribute during runtime are reflected in the value of the segment.

If the display type of the segment is a check box, you can set whether the default value of the segment is checked or unchecked.

Define Attachments

Attachments: Explained

Attachments are pieces of supplementary information that users can associate with specific business objects such as expense reports or purchase orders. Attachments can be URLs, desktop files, text, or in cases where available, repository folders. For any given business object, a user may be able to only view attachments, or also create, delete, or edit attachments, depending on security. For more information on an introduction to attachments, see the Oracle Fusion Applications Developer's Guide.

Repository

Attachments are stored in a content 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.

Action	Privilege
View	Read Application Attachment (FND_READ_APPLICATION_ATTACHMENT_DATA)
Add or Update	Update Application Attachment (FND_UPDATE_APPLICATION_ATTACHMENT_DATA)
Delete	Delete Application Attachment (FND_DELETE_APPLICATION_ATTACHMENT_DATA)

For example, if users have the Read Application Attachment privilege for all categories associated with the expense report attachment entity, except the Receipts attachment category, then they can view all expense report attachments

except those created with the Receipts category. Likewise, if users do not have the Update Application Attachment privilege for any attachment categories tied to the expense report attachment entity, then they cannot create any attachments at all for expense reports.

For more information on attachment category data security, see the Oracle Fusion Applications Developer's Guide.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

Issue: Missing Attachment Category

Users can see existing attachments, but the attachments no longer have an attachment category value.

Resolution

When the attachment was added, at least one category existed for the corresponding attachment entity, as otherwise the attachment could not have been added. Since then, the entity was edited so that it no longer has any assigned categories, so the user cannot see the category associated with that attachment.

Use the Manage Attachment Entities page to reassign attachment categories to the relevant attachment entity. For example, if users can no longer see the Receipts attachment category for an attachment to an expense report, then search for the expense report attachment entity and assign to it the Receipts category. You might need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or what additional categories to assign.

Finally, certain attachments UI for users have predefined restrictions on categories in place. Your developers can also introduce additional filters to determine which document categories are available for a specific page. Check with your developers or help desk.

FAQs for Define Attachments

What's an attachment category?

An attachment category is used to classify and secure attachments. Each attachment user interface must be defined with at least one category for users to be able to add attachments. If there are multiple categories, users can view them and select one when adding attachments. For example, attachments for an expense report can be categorized as receipts, scanned invoice images, and so on.

You can also associate roles with categories to determine user access and actions for attachments, based on the categories assigned to the attachment entity. For example, security for expense report attachments can be based in part on the categories assigned to the expense report attachment entity. You can define multiple categories per module, and add and manage custom categories for your own purposes. For more information on attachment category data security, see the Oracle Fusion Applications Developer's Guide.

Use the Manage Attachment Categories page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Attachment Categories task.

Set Activity Stream Options

Activity Stream Options: Highlights

Activity Stream is a region on the Oracle Fusion Applications Welcome dashboard and other pages in various applications. Users track the activities and transactions of other users in this region. You can set options that affect the all Activity Stream regions for all users across your site. Individual users can still override your settings through Activity Stream preferences.

Activity stream settings are described in the Oracle Fusion Middleware User's Guide for Oracle WebCenter Portal: Spaces. When you read content from that guide, note that:

- Your setup applies to all users, not just yourself or any individual user.
- You can disregard discussions about how to access the settings, because you access the Set Activity Stream Options page by starting in the Setup and Maintenance Overview page and searching for the Set Activity Stream Options task.

Setting Activity Stream Options

- Define the types of users to display activities about in the Activity Stream region, the types of activities to track, and other settings.

See: Setting Activity Stream Preferences

Manage Menu Customizations

Managing Menu Customizations: Highlights

You can customize the Navigator menu, which is the main menu of Oracle Fusion Applications and is always available in the global area. You can also customize the home page of Oracle Fusion Applications by adding tabs to display additional dashboards. Use the Manage Menu Customizations page, which you can access by starting in the Setup and Maintenance Overview page and searching for the Manage Menu Customization task.

An overview of customizing the Navigator menu and home page is provided in the Oracle Fusion Applications Extensibility Guide.

Navigator Menu and Home Page

- Navigator menu customization involves managing items, which are nodes in the menu that take the user to the desired destination, and groups, which are categories of items.

See: Customizing the Navigator Menu

- The Oracle Fusion Applications home page displays a set of dashboards as tabs. You can create more tabs and make them display additional dashboard content.

See: Customizing the Navigator Menu.

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.

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.

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>.

Oracle Fusion Middleware Products

- Configure business objects to enable auditing in Oracle Fusion Middleware products. Refer to the Oracle Fusion Middleware Security and Administrator's Guide for Web Services.

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 Oracle Social Network to collaborate with key individuals within your organization, as well as external contacts such as suppliers and partners, using critical transaction data from Oracle Fusion Applications. Available collaboration tools include groups, activity feeds, discussion forums, document sharing, instant messaging, email and web conferencing. By bringing key attributes from the applications to share, socialize and update information, you can make better business decisions, based on additional information that you obtain and analyze within your social network environment. You do this by defining which business objects and attributes to enable and the enablement method for Oracle Social Network integration with Oracle Fusion Applications. Once enabled, these business objects can be shared in collaborative conversations. Within Oracle Fusion Applications, conversations can be viewed using the Social button in the Global Area and select applications pages.

Use the Manage Oracle Social Network Objects page, which you can access by starting in the Setup and Maintenance Overview page and searching for the task Manage Oracle Social Network Objects.

Note

Oracle Social Network is currently not available in non-Cloud implementations.

Aspects of managing Oracle Social Network objects include the following:

- Enabling business objects for integration.
- Understanding initial and subsequent defaults.

Enabling Business Objects for Integration

A business object cannot be shared within Oracle Social Network until a functional administrator or implementor accesses the Manage Oracle Social

Network Objects page and enables the business object for Oracle Social Network integration. Use the Manage Oracle Social Network Objects page to set up and define which business objects and attributes to enable and the enablement method for Oracle Social Network integration with Oracle Fusion Applications.

Note

You see only the business objects that are predefined as available to enable, not the full list of business objects for products that you are permitted to view.

Understanding Initial and Subsequent Defaults

Upon initial deployment, the preceding rule that states a business object cannot be shared until it is accessed and enabled through the Manage Oracle Social Network Objects page, does not apply to certain default configurations for business objects in Oracle Fusion Customer Relationship Management (CRM) and Oracle Fusion Human Capital Management (HCM). Subsequently, these business objects are managed no differently than any of the other business objects. If you set to Manual or Automatic, the setup is not complete until you select attributes.

Options for Enabling Oracle Social Network Objects: Explained

Access the Manage Oracle Social Network Objects page to select from the following options to enable business objects and apply attributes for Oracle Social Network integration with Oracle Fusion Applications.

These enablement options exist:

- Enable Object
- Disable Object
- Enable All

Enable Object

Access the Business Object section of the Manage Oracle Social Network Objects page.

Click **Enable Object** to select from the available enablement options and define how the object integrates with Oracle Social Network. Options include:

- **Manual** - Recommended; the object is enabled for manual sharing with Oracle Social Network, where the social network user decides whether or not to share each instance of the object with Oracle Social Network. Once shared, all updates to enabled attributes of the object instance, and deletes, are sent to Oracle Social Network. Updates to attributes that are not enabled are not sent.
- **Automatic** - Automatically sends newly enabled object instances and updates to Oracle Social Network. All object instances are automatically shared with Oracle Social Network upon creation and all subsequent updates to enabled attributes of the object instances, and deletes, are automatically sent to Oracle Social Network.

- **No** - Does not send any object instance information to Oracle Social Network.

Note

After you enable an object, you must enable one or more attributes in the Attributes section of the Manage Oracle Social Network Objects page. Updates to enabled attributes are sent to Oracle Social Network.

Disable Object

Access the Business Object section of the Manage Oracle Social Network Objects page.

Click **Disable Object** to disable the selected business object, then Save. This shortcut disables the currently selected business object by changing the enablement option to No.

Alternatively, you can click **Enable Object** to access the dialog box and enablement options and select **No**, then click OK, and then click Save to disable a selected business object.

Disabling the object through either mechanism does not affect any associated attributes. Use the Attributes section of the page to manage the list of enabled attributes.

Enable All

Access the Manage Oracle Social Network Objects page.

Click **Enable All** at the page level to perform a bulk update of the enablement status from No to Manual of all business objects. After you click Enable All, then you save.

The bulk update of the statuses does not affect the attributes selected as enabled for each business object. Use the Attributes section of the page to manage the list of enabled attributes.

Status Column

The Status column appears in the Business Objects table to provide a visual indicator as to which enabled business objects do not yet have an enabled attribute assigned.

When you enable a business object (automatic or manual), a check mark or a warning sign appears in the Status column of the business object depending on this criteria:

- A check mark indicates that you have configured attributes for an enabled business object.
- A warning sign indicates that you have not configured any attributes for the enabled business object.

Note

It is important that you configure the attributes before you click Save.

When you enable a business object, but do not configure any attributes for the enabled business object, unexpected results can occur as no attributes are sent to Oracle Social Network during create and update except some internal bookkeeping information. Deletes are sent as usual.

Attributes

Enable business object attributes in the Attributes section of the Manage Oracle Social Objects page. To provide a visual indicator of the enabled business objects that have no attributes added, a Status column appears at the end of the Business Objects table. This column shows a warning when there are no attributes defined for a business object that has had its enabled state changed from No. This warning does not prevent you from saving the Oracle Social Network enablement of the business object, but does provide a guide that you should assign attributes for an enabled business object.

In the Attributes table, you can do the following:

- View a list of all attributes that are enabled.

These are the attributes that, upon update, are sent to Oracle Social Network. Disabled attributes are not sent.
- Click Add to launch the dialog where you can select attributes to add to the table.
- Click Remove to remove attributes from the table.

FAQs for Manage Oracle Social Network Objects

What happens if I update translations?

When you click **Update Translations** from the Manage Oracle Social Network Objects page, you synchronize the newly translated text from Oracle Fusion Applications so that it can be used within Oracle Social Network. This means you can install and enable a new language or take a language patch at any time.

Update Translations sends attribute labels and business object names to Oracle Social Network for use in its user interface. Users in Oracle Social Network see the attribute or business object labels in the language of their locale. If the user changes locale in Oracle Social Network, then the attribute or business object labels appear in the updated language. The data, however, appears in the language in which it was originally sent to Oracle Social Network. If a user has previously sent a business object instance to Oracle Social Network, then the business object instance data is not updated. When you save the business object enablement to Oracle Social Network, it sends the translations as well. This makes it unnecessary to click both Save and Update Translations. For subsequent updates to labels and attributes, such as if a new Oracle Fusion Applications language pack is installed, run Update Translations to send the updated labels and their translations to Oracle Social Network. The Update Translations button sends translations for all business objects whose enablement is Manual or Automatic.

Manage Applications Core Common Reference Objects: Manage Applications Core Messages

Common Messages: Points to Consider

Common messages, which have message names that begin with FND_CMN and message numbers between 0 and 999, are used throughout Oracle Fusion Applications. Each common message can appear in multiple places in any product family. For example, the FND_CMN_NEW_SRCH message can be used for any search to indicate that no results were found. Common messages that are of type error or warning are part of the message dictionary.

Editing Common Messages

Because a common message can be used in any application, consider the ramifications if you edit any aspect of the message, including incident and logging settings. Changes would be reflected in all instances where the message is used. For example, if you change the message text, make sure that the text would make sense to all users across Oracle Fusion Applications who might see it.

Creating Common Messages

You can create custom common messages for use in multiple places within a single product. Do not begin the message name with FND_CMN, but use another suitable convention. The message number should be within the range that is designated for the product.

Manage Applications Core Common Reference Objects: 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.

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 Fusion 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.

Common Applications Configuration: Define WebLogic Communication Services Configuration

Click-to-Dial: Explained

Use Click-to-Dial to place a call to a contact from a hyperlink on the phone number or phone icon.

Here are a few topics that are important to know when using Click-to-Dial:

- Normal call flow
- Interaction Records and Notes
- Operational Notes

Note

Click-to-Dial must be enabled to make calls using the various contact information pages and pop-up UIs. When enabled, phone numbers appear as hyperlinks. Interaction logging is available if that feature is enabled. If interaction logging is available, a note indicating that fact will be displayed in the UI. See Click to Dial: Top Tasks for more information about enabling these features.

Normal Call Flow

Click-to-Dial uses a call-the-caller-then-call-the-callee procedure for completing a phone call. That format and the normal flow of this procedure are described below.

- **You initiate a call**

If you see a small orange square next to a contact or customer name, click the square to display further details, including phone numbers. To place a call, place your mouse over the phone number hyperlink and click.

Note

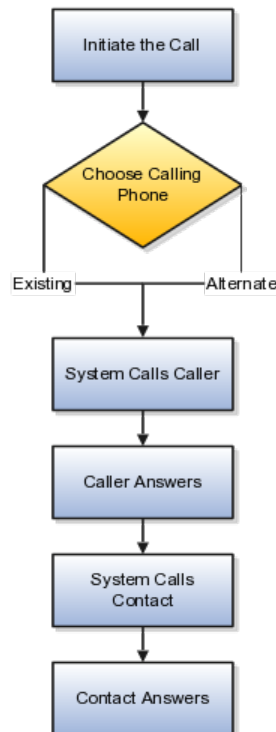
Click-to-Dial does not work on phone numbers that are marked with a Do Not Call icon.

-
- **Select a Calling Phone**

Choose the calling phone number. Usually the calling phone is a number from your profile information. Alternately, if you need to use a phone not in your profile, you can specify a different number to originate your call.

- **Call Flow**

After you select the calling phone number, the system calls you back on that number, waits for you to answer, and then calls the person for whom the call is intended.



Interaction Records and Notes

Click-to-Dial automatically creates an interaction record of the call, when that feature is enabled. The details window that provides the phone number may also show an Interaction icon that you can click to display a list of interaction records to edit, for example to provide a description of the call. The window may also provide a notes feature that you can use to record notes during the call.

Interaction Logging

The interaction record is logged as soon as the call is either successfully set up or known to have failed.

The interaction log records the customer, call participants, a timestamp noting the start time of the call, the direction of the communication, in or outbound, and

the resolution code. The description is automatically updated with these three items:

- Call ID from OWLCS
- Your chosen phone number
- Contact phone number

The call resolution code is determined from OWLCS and recorded in the interaction:

OWLCS Call Status	Resolution Code in Interaction
CallConnected	CALL ANSWERED
CallAborted	FAILED
CallHangUp	FAILED
CalledPartyBusy	NOT AVAILABLE
CalledPartyNoAnswer	NOT AVAILABLE
CalledPartyNotReachable	NOT AVAILABLE
CallingPartyBusy	FAILED
CallingPartyNoAnswer	FAILED
CallingPartyNotReachable	FAILED

Editing interactions

Once the call is established, if Interactions is available, you can use the Interactions icon on the UI to launch the interaction record list view. Select the current interaction record to edit it.

Operational Notes

Because of the call-the-caller-then-call-the-callee format, there are some conditions that may occur due to several calling situations. Some of these conditions are described below:

- Why don't I hear a ring-back tone? As soon as you answer the system call-back, the system immediately dials the contact. You won't hear a ring-back tone as in a normal outbound phone call. However, you can tell that the call attempt is progressing because:
 - The phone indicates that the connection is active. If the call to the contact reaches a busy tone or the call attempt times out, the connection is dropped.
 - The dialing window stays on the screen while the call attempt is progressing. It disappears when the connection is either successfully established or fails.
- What if your phone is busy and the call-back goes directly to voice mail? Normally this would not happen because you would not initiate a new call when you are already busy on another call. However, this situation could occur due to a race condition, that is where another incoming call reaches your phone before the Click-to-Dial call-back. When this happens, two different scenarios could occur:

- If your phone is configured for busy-forward-all-to-voice-mail, the Click-to-Dial call would be forwarded to your voice mail, and the system thinks that the caller has answered the call and will proceed to call the contact. On answering, the contact hears your voice-mail greeting.
- If your phone is capable of presenting a second call to the user, as is supported by many office phones and mobile phones, then you can still answer the Click-to-Dial call and there is no issue.
- What if you wait too long to answer the call-back? In other words, you wait longer than the ring-no-answer-forward-to-voice-mail timer on the phone system and the call goes to voice mail. Normally, this would not happen because you are expecting the inbound call after you started the call, and would answer promptly. However, if for some reason you do not answer and allow the call to ring-no-answer-forward to voice mail, then the system would think that you have answered the call and will proceed to call the contact. On answering, the contact hears your voice-mail greeting.
- What if the contact does not answer in 30 seconds and the system abandons the call attempt? If the contact's voice mail is configured to answer after 30 seconds, you will not be able to leave a message.

Click-to-Dial: Top Tasks

Click-to-Dial is a feature of the customer contact process. Phone communication to customers and employees is initiated with a click of the mouse, leveraging your customer contact information and the application context. The Click-to-Dial feature uses Oracle WebLogic Communication Services, OWLCS, to enable communications. Applications that provide the Click-to-Dial functionality do so primarily through contextual actions in the application.

Additionally, Click-to-Dial utilizes Oracle Fusion Interactions as an optional transaction logging feature that will track information about the call such as the customer, call participants, a timestamp noting the start time of the call, the direction of the communication, in or outbound, and the resolution code.

Click-to-Dial integrates with your telephony environment and must be manually enabled in your deployment. This topic highlights what is required to set up the Click-to-Dial feature and to implement logging of the calls made using the Click-to-Dial feature.

Terms used in setting up these communications

- PSTN: Public switched telephone network is the network of the world's public circuit-switched telephone networks.
- SIP: Session initiation protocol, an open signaling protocol standard that is used to set up phone calls
- TPCC: Third Party Call Control enables an application to control the telephony network to set up calls automatically.
- OWLCS: Oracle WebLogic Communication Services. Offers the TPCC service to Fusion applications and sets up the calls via SIP integration with the telephony network.

The set up task list Define WebLogic Communication Services Configuration delineates four tasks required for the correct configuration and implementation of Click-to-Dial. There is an optional task, separate from the set up task list, required for implementing Interaction logging.

Information about implementing Click-to-Dial can be found in the Oracle Fusion Applications Administrator's Guide. Detailed information about configuring and maintaining WebLogic Communication Services is found in the Oracle WebLogic Communication Services Administrator's Guide

Configure and Deploy WebLogic Server

- Deploy WebLogic Communication Services: After the Oracle WebLogic communication server is deployed, this manual task activates the server.

See: Oracle WebLogic Communication Services Administrator's Guide

Integrate Communications Services

- Integrate WebLogic Communication Services with Telephony Network: This manual task integrates communications within the telephony environment. OWLCS must be configured to interface with the specific characteristics of the telephony network.

See: Managing Oracle WebLogic Communication Services for Click-to-Dial Functionality

Specify the Domain and Address

- Register a URL for the telephony gateway or soft switch for SIP domain: This task defines the Server protocol, defaulted to http, the external server host address and external server port address. The Associated Modules section is not required for setup. You can also perform this as a manual task using the Oracle Fusion Topology Manager to configure the address of the SIP Public Switched Telephone Network (PSTN) gateway or SIP soft switch serving the users within that domain. This address is needed by Click-to-Dial to correctly form the SIP addresses required by WebLogic Communication Services. See the link to Configuring PSTN Gateway Address Using Topology Manager: Worked Example.

Enable Click-to-Dial

- After configuring the server and defining the SIP domain, perform the Enable Click-to-Dial task. This task sets the value of the profile option Enable Click-to-Dial to 'Yes.'

Call Logging via Fusion Interactions

- To initiate the Interaction based logging for Click-to-Dial, set the profile option Call Interaction Logging Enabled to 'YES.'

Configuring PSTN Gateway Address Using Topology Manager: Worked Example

This example demonstrates how, during the implementation of the **Register URL for the telephony gateway or soft switch for SIP domain** task,

you must manually configure the PSTN gateway address by registering HzCTDPstnGatewayApp to a given environment using Oracle Fusion Topology Registration

These steps configure the address of the SIP Public Switched Telephone Network (PSTN) gateway or SIP soft switch serving the users within that domain. This address is needed by Click-to-Dial to correctly form the SIP addresses required by WebLogic Communication Services.

For example: SIP:+1650-555-1212@pstn_gateway.oracle.com;user=phone where pstn_gateway.oracle.com is the SIP domain. The SIP domain can also be expressed in the format 10.1.1.1 (IP address).

Configuring PSTN using the Topology Manager

1. Sign in to Oracle Fusion Applications as a user that has **application implementation consultant** and **WebLogic Services administration** roles
2. In Fusion Applications Setup and Maintenance, click Register Enterprise Applications from the regional area under **Topology Registration**
3. On the Register Enterprise Applications page, click the plus icon to add an enterprise application. An Add Enterprise Application popup appears
4. Enter the new application information: Click Search in the Enterprise Application list field. Enter **HzCTDPstnGatewayApp** in the name field and click Search. Click OK.
5. Enter the other fields in the Add Enterprise Application popup

Field	Value
Name	HzCTDPstnGatewayApp
Server Protocol	SIP This field is ignored by click-to-dial. Oracle WebLogic Communication Service (OWLCS) always uses the SIP protocol.
External Server Host	10.143.167.91 (Used as an example) A host name can be used instead of an IP address.
External Server Port	0 (Used as an example) This field is ignored by Click-to-Dial.

6. Click Save and Close

Common Customer Data Management Configuration: Define Source Systems

Source Systems: Explained

You can set up source systems to enable users to identify the source of the data they are importing into the Oracle Fusion database. You can specify whether the source system is a Spoke system, such as a legacy system, or a Purchased system, such as data from a third party provider. You can also specify what types of entities the source system contains, for example, you can specify that a source system will contain trading community member data.

You can configure the following for a source system:

- Source system code, name, and description
- Source system type
- Enable for Items, Trading Community Members, Order Orchestration and Planning, and Assets

Source System Code, Name, and Description

You can create a source system code to uniquely identify the source system. Source system codes are used by the application to create references between source IDs and the Oracle Fusion Applications database IDs. You can create a source system name and description to provide information that is more descriptive than the source system code.

Note

You cannot update the source system code once you have created the source system.

Source System Type

You must set up a source system as either a Spoke system, such as a legacy system, or a Purchased system, such as data from Dun & Bradstreet.

Enable for Items, Trading Community Members, Order Orchestration and Planning, and Assets

You should select which types of entities will be imported from the source system into the Oracle Fusion Applications database from the following:

- Items
- Trading Community Members
- Order Orchestration and Planning
- Assets

You can select one or more of these entity types as required for the source system. It is important to enable the correct entity types because each import UI filters source systems based on their entity type. For example, if a source system is enabled for Trading Community Members, Items, and Assets, then the source system can be selected as a data source in the Trading Community Members, Items, and Asset import UIs; however, the source system won't be able to be selected in the Orchestration and Planning import UI.

Source System Entities: Explained

Source System Entities are the entities, such as addresses and parties, which can be imported using a specified source system.

When you import data from a source system, all of the entities in the source system data will be imported. Within the Source System Entities UI, you can choose to allow multiple source references, which allows multiple records from a source system to map to a single trading community record.

FAQs for Define Source Systems

What happens if I allow multiple source system references?

Allowing multiple source system references means that when you import data from a source system you can merge multiple, or duplicate, source system records and create one record in the Oracle Fusion Applications database.

If you do not allow multiple source system references then an Oracle Fusion Applications database record will be created for every source system record. This means that you could potentially create duplicate records in the Oracle Fusion Applications database.

Common Customer Data Management Configuration: Define Party Usages

Party Usage: Explained

Party usages describe how a party is used in the context of the implementing organization. For example, a person in the business community may be a consumer or contact. Rules can be associated to a party, and these rules determine when a party usage can and cannot be used.

These are the following rules that can be defined for a party usage:

- Assignment rules
- Exclusivity rules
- Incompatibility rules
- Transition rules

Assignment rules

Assignment rules are used to define how the party usage can and cannot be assigned to parties. You can choose to allow unconditional party usage assignment, enabling the assignment to be manually created, or updated, by an administrator.

Alternatively, you can restrict the manual assignment and update of a party usage, which will mean that the party usage can only be created, or updated, by a business event. For example, creating an account for a party record can automatically assign the Customer party usage to that record.

You can also specify whether the party usage assignment can also be created or updated by the assignment of a party relationship. For example, the party relationship **Contact Of** can assign a party usage of Organization Contact.

Exclusivity rules

Exclusivity rules enable you to restrict party usage assignment, so that between a specified date range the party usage can be the only usage allowed to be

assigned to a party. For example, you can set up an exclusivity rule that between January 1 2011 and February 1 2011 parties that have the Manufacturer party usage cannot have any other party usage assigned to them.

Incompatibility rules

You can specify which party usages cannot be assigned concurrently to a party between a defined date range. For example, you can specify that you cannot assign a Sales Account usage to a party with a Sales Prospect usage, during January 1 2011 and December 31 2030.

Transition rules

You are able to define which party usages can transition to the party usage you are creating or editing. Once a party has transitioned to the current party usage, the previous party usage is set with an end date. For example, when a party with the Prospective Partner party usage is assigned the Partner party usage, the Prospective Partner usage is set with an end date.

Party Usage Filter Rules Explained

Party usage filter rules are a grouping of party usages that are used for searching and filtering in user interfaces. You can specify which party usages you want to include and exclude, and whether the party usage assignments are active or inactive.

For example, an Eligible Customers filter rule can include the Sales Prospect party usage, Sales Account, Legal Entity, and Customer party usages. The Eligible Customers filter rule can then be used in transactional UIs to filter out parties with party usages that are not included in the rule.

Note

Once a party usage filter rule is created you need to compile the filter rule before you can use the filter rule in a User Interface.

FAQs for Define Party Usages

How can I compile a party usage filter rule?

In the Party Usage Filter Rule page click **Actions** in the header section, and then click **Compile Filter Rule**.

Common Customer Data Management Configuration: Define Names

Additional Name Types: Explained

Define additional name types to capture alternative names for parties.

For example, create an additional name type NICK_NAME to capture the nick names of parties of type Person.

While creating an additional name type, specify the party type to which it applies. From then on, the additional name type will be available as a naming option for all parties of that party type within the deploying company. You can use additional name types to capture language-specific names as well.

Common Customer Data Management Configuration: Define Identifiers

Identifier Types: Explained

Create additional identifier types to provide extensions to party attributes. For example, you can create an additional identifier type to record a person's passport number.

You can choose which party types can use the additional identifier type. These party types can be Person or Organization, or both. You can also specify whether the value of an identifier type must be unique. For example, the passport number listed under each person's profile must be unique.

Additional identifier types do not automatically appear in the user interface. If you want to use identifier types in the application you will need to call the **Trading Community Member Name and Identifier Setup** web service.

Common Customer Data Management Configuration: Define Relationships

Relationships Types: Explained

A relationship type categorizes relationship phrases and roles that determine the nature and purpose of a relationship. A relationship type defines the aspects of a relationship such as the roles of the subject and object, business rules, and phrases that describe the relationship. A relationship, uses the relationship type, to define how a party is related to another party.

The components of a relationship type are:

- Relationship phrase pair and role pair
- Relationship characteristics

Relationship phrase pair and role pair

A relationship type contains a relationship phrase pair and a role pair. Relationship phrase pair and role pair define the mutual roles that the two parties play in a relationship. For example, for relationship type Employment the phrase pair contains Employee of and Employer of; and, the role pair contains Employee and Employer.

The role pair describes the roles of the entities immaterial of the direction of the relationship. For example, in an Employment relationship Bill has the role of an employee and Oracle Corporation has the role of an employer.

The phrase pair describes the roles of the entities in the relationship. For example, in an Employment relationship if Bill is the subject and Oracle is the object, then the phrase would be Employee of. Similarly, if Oracle is the subject and Bill is the object, then the phrase would be Employer of.

Relationship characteristics

Relationship types include characteristics or rules that define how the relationship type can be used.

While defining a relationship type, you can define if the relationship can be:

- Circular

- Related to self

Circular relationships have two parties related in a circle of relationships. For example, you can create a Competitor relationship type to link two organization parties. In this scenario, ABC Corp is a competitor to XYZ Corp, and XYZ Corp is also a competitor to ABC Corp.

You can create a self-relating relationship type, where the subject and the object of the relationship are the same. For example, you can create a Subsidiary relationship type, where a subsidiary organization ABC is linked to a parent organization XYZ. The parent organization XYZ, in turn, could be linked to itself as a subsidiary using the same relationship type.

Relationship Groups: Explained

Relationship groups categorize relationship phrase and role pairs for functional purpose. For example, you can assign the relationship phrase Subsidiary Of to a group, and assign the phrase Headquarters Of to a different group.

Uses of Relationship Groups

Relationship groups are used to determine which relationship roles and phrases are displayed in specific user interfaces of the Oracle applications. A relationship group can correspond to one or more Oracle Applications user interfaces. For example, you can configure an Oracle application to display only the relationship types that pertain to the Party Contacts information.

Manage Relationship Types

Creating Relationship Types: Worked Example

This example demonstrates how to create a relationship type, and add relationship roles and phrases to the relationship type.

The following table summarizes key decisions while relationship types.

Decisions to Consider	In This Example
What are the subject and object party types for the relationship?	The subject type and object type is organization.
Is the relationship circular?	Yes
Can the subject related to self in this relationship?	No

You must search for the relationship type to make sure it does not already exist, before you create a relationship type.

Searching for a relationship type

1. From the Setup and Maintenance work area, search for the Manage Relationship Types task and click **Go to Task**.
2. On the Manage Relationship Types page, in the Search region, complete the fields, as shown in this table.

Field	Operator	Value
Relationship Type	Equals	Vendor
Subject Type	Equals	Organization
Object Type	Equals	Organization

3. Click **Search**.

Review the search results. Ensure that there is no existing Relationship Type for the criteria.

Creating a relationship type

1. In the Search Results region, click the **Create** icon to open the Create Relationship Type page.
2. On the Create Relationship Type page, in the Overview region, enter Vendor in the **Relationship Type** field.
3. Select the **Circular** check box.
4. In the Relationship Roles and Phrases region, click the **Create** icon to open the Create Relationship Role and Phrase Pair dialog box. When you create a relationship with a relationship phrase or role, the reverse direction is automatically created with the other phrase or role in the pair.
5. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

Field	Value
Subject Type	Organization
Subject Phrase	Vendor of
Subject Role Singular	Vendor
Subject Role Plural	Vendors
Object Type	Organization
Object Phrase	Vendee of
Object Role Singular	Vendee
Object Role Plural	Vendees

6. Click **OK**.
7. In the Create Relationship page, click **Save and Close**.

Editing Relationship Types: Worked Example

This example demonstrates how to edit a relationship type, including adding relationship roles and phrases to the relationship type, and assigning the relationship to a group.

In this example, you will add a role and phrase pair to the relationship type Employment. You will also add the Employment relationship type to the Party Contacts group.

Editing a relationship type

1. From the Setup and Maintenance work area, search for the Manage Relationship Types task and click **Go to Task**.
2. On the Manage Relationship Types page, in the Search region, enter Employment in the **Relationship Type** field.
3. Click **Search**.
4. In the Search Results region, select the Employment relationship type.
5. Click the **Edit** icon to open the Edit Relationship Type page.
6. On the Edit Relationship Type page, in the Relationship Roles and Phrases region, click the **Create** icon to open the Create Relationship Role and Phrase Pair dialog box.
7. Complete the fields, in the Create Relationship Role and Phrase Pair dialog box, as shown in this table.

Field	Value
Subject Type	Organization
Subject Phrase	Has contract employees
Subject Role Singular	Contractor
Subject Role Plural	Contractors
Object Type	Person
Object Phrase	Contract Employee of
Object Role Singular	Contract Employee
Object Role Plural	Contract Employees

8. Click **OK**.
9. In the Relationship Groups region, click the **Add Row** icon.
10. In the **Name** drop-down list, select Party Contacts.
11. Click **Save and Close**.

Manage Relationship Groups

Creating Relationship Groups: Worked Example

This example demonstrates how to create a relationship group. Relationship groups are used to categorize relationship roles and phrases, based on the user interfaces they appear in.

In this scenario, you will create group for relationship types that link an organization to an external organization.

Creating a relationship group

1. From the Setup and Maintenance work area, search for the Manage Relationship Groups task and click **Go to Task**.
2. On the Manage Relationship Groups page, in the Search Results region, click the **Create** icon to open the Create Relationship Group page.
3. Complete the fields, as shown in this table:

Field	Value
Name	Related External Organizations
Code	REL_EXT_ORGS
Description	Indicates a group of relationships that link the organization to external entities.

4. In the Relationship Roles and Phrases region, click on the **Add Row** icon to add a relationship role and phrase pair.
5. For each relationship role and phrase pair, complete the fields, as shown in this table.

Subject Type	Subject Role
Organization	Supplier
Organization	Vendor
Organization	Competitor

6. Click **Save and Close**.

Common Customer Data Management Configuration: Define Classifications

Classifications: Explained

The classifications model provides you with a flexible tool to categorize entities such as parties, projects, tasks, and orders. Classifications enable you to classify an entity, such as a party, in a way that the rest of the world sees it, in addition to the way that it is referenced within your organization.

The major components of classifications are:

- Classification categories
- Classification rules
- Classification codes
- Classification code hierarchy
- Entity assignment

Classification Categories

Classification categories give you the ability to classify entities under a broad subject area. For example, you can classify organizations based on the industries they operate in. Classification categories are a logical grouping of one or more classification codes and allow classification code rules to be defined.

Classification Rules

Classification categories can have rules that define how classifications can be assigned to entities. When you set up classification categories specific rules can be created, such as allowing the parent classification code to be assigned to a party, and allowing multiple classification codes to be assigned to an entity.

Classification Codes

The individual values within the classification category are called classification codes. For example, in the 1987 SIC classification category there is a classification

code of software that can be assigned to a party in the software industry. You can organize classification codes into a hierarchical tree, with a parent classification code at the top of the tree and child classification codes branching off from the parent code or other classification codes.

Classification Code Hierarchy

You can create hierarchies of classification codes within a classification category. For example, you can set up a classification category of IT containing the classification codes hardware, keyboards, and printers. You can then set up the classification code of hardware as the parent code at the top of the tree, with the classification codes of keyboards and printers as child codes underneath. You can create further child classification codes, such as dot matrix, ink-jet, and laser below the printer classification code.

Entity Assignment

Define which entities can be assigned to a classification category by entering the entity table name and creating a Where clause in SQL. Only entities that satisfy the Where clause are assigned the classification category. For example, a classification category called industries with the Where clause of where "party_type = ORGANIZATION" would have the result that only organizations can be classified with the industries classification category.

FAQs for Define Classifications

What happens if I allow parent code assignment?

You can assign the parent classification code to an object, as well as the child classification codes. The parent classification code is the code at the top of the classification code tree.

If you don't allow parent classification codes to be assigned to an object, then you can assign only child classification codes, or codes that are below another classification code in the tree, to an object.

What happens if I allow multiple class code assignment?

You can assign more than one classification code from this classification category to an object.

If you don't allow multiple classification codes to be assigned to an object, then you can assign only one classification code from this classification category to an object.

Can I update an existing entity assignment rule?

No. You can delete the entity assignment rule and create a new one.

Common Customer Data Management Configuration: Define Data Formats

Data Formats and Setup Activities : How They Fit Together

Data Formats help you define address formats and name formats within your organization, and assign them to specific geographies. These data formats can be used to capture and validate party name and address information. In order to use geography-based validations for addresses, you must set up geography data in addition to the address formats.

Managing Data Styles

You can define new styles for name and address, which would categorize a name and address style format. A Data Style, such as Mailing Address or Concatenated Name, categorizes name and address style formats. Oracle Fusion uses Data Styles to identify a particular manner or situation for which an entity, such as name or address, is formatted. The application can be customized so that, for example, only certain name and address styles are displayed in the user interface.

How Data Formats ensure Data Quality

Use Data Formats to ensure party address and party name data quality. The format is used to present the address elements in the country specific formatting, and help users during address entry. This also provides the flexibility to specify the scope of address validation. Address formats are linked to the geographic data in order to provide address verification, at the data-import level. For example, when creating or editing an address for a party, selecting a country displays the address format for that country.

The use of Name formats ensures that name components are stored in a decomposed manner. This allows you to construct name representations from components in a variety of different styles. Name formats also allow you to store phonetic names that are required for certain geographies.

Data Formats simplify the storage of party address and party name, by storing them as components. Once you setup the address and name format for a geography, you can use the format to validate address and name data.

Setup Formats: Explained

Data Formats determine how names and addresses are displayed in the Oracle Fusion application. Name and address formats vary depending on usage, country, and language. You can create name formats and address formats, customize them to display the address and name components, and assign them to specific locales. This allows you to create data formats for various countries, languages, and usages. Additionally, you can create variations of the formats, with each variation having its own layout. For example, an address format may have variations of a general address, rural address, and military address.

There are two types of data formats.

- Address format
- Name format

Address Formats

Address formats specify the layout of an address, including the address components and their position in the layout. For example, an address format for US postal address can include address, city, state, and zip code. Similarly, the address format for UK would include address, city, county, and post code.

Name Formats

Name formats, such as Informal or Mailing, allow you to specify how a name is displayed in the application, based on the usage, language, and country. A name format includes name elements, their position, and formatting. You can create a name style format for either an organization name or a person name.

Creating an Address Style Format: Worked Example

This example shows how to create an address style format for a specified address style.

Specify the address style format name and address style, create the format variation layout, and assign a locale.

Creating the address style format name and specifying the address style

Create the address style format code and name, choose which address style you want to use to define this address style format, and specify if you want this address style format to be the default format for the selected address style.

1. On the Manage Address Formats page, click **Actions** and then click **Create**.

2. On the Create Address Style Format page, complete the fields in the Overview section, as shown in this table:

Field	Value
Code	CA_POSTAL_ADDR
Name	Canadian Postal Address Format
Address Style	Postal Address
Default	No

3. Click **Save**.

Creating the format variation layout

Specify the address lines you want in the address and the position of the lines in the address layout. You also want to specify if the address lines will be mandatory, whether the address lines will render in uppercase, and whether there will be blank lines after the address line.

1. In the Format section on the Create Address Style Format page, click on the **Format Layout** tab.
2. In the Format Variation section, click **Actions** and then click **New**.
3. Enter a variation number in the **Variation** field.
4. In the Format Variation Layout section, click **Actions** and then click **New**.
5. Complete the fields as shown in this table:

Field	Value
Line	1
Position	1
Prompt	Address line 1
Address Element	Address line 1
Required	Yes
Uppercase	No

6. Click **Actions**, then click **New** to create another address line.
7. Complete the fields as shown in this table:

Field	Value
Line	2
Position	2
Prompt	City
Address Element	City
Required	Yes
Uppercase	Yes

8. Click **Expand** on the City address line, then enter the value **1** in **Blank Lines Before**.
9. Continue to create as many address lines as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, an address line. In this section you can also create an Attribute Transform Function to change a data value into a different value.

Assigning a locale

Specify the relevant countries for this address style format.

1. In the Format section on the Create Address Style Format page, click on the **Locale Assignment** tab.
2. Click **Actions** and then click **New**.
3. Choose a country for the address style format.
4. Continue to add the countries that will use this address style format.
5. Click **Save and Close**.

Creating a Name Style Format: Worked Example

This example shows how to create a name style format for a specified name style.

Specify the name of the name style format and name style, create the format variation layout, and assign a locale.

Creating the name of the name style format and specifying the name style

You create the name style format code and name, and choose which name style you want to use to define this name style format. You also specify if you want this name style format to be the default format for the selected name style.

1. On the Manage Name Formats page, click **Actions** and then click **Create**.
2. On the Create Name Style Format page, complete the fields in the Overview section, as shown in this table:

Field	Value
Code	CONCAT_NAME_US
Name	United States Concatenated Name Format
Name Style	Concatenated Name
Default	No

3. Click **Save**.

Creating the format variation layout

Specify the name parts you want in the name, and the position of the name parts in the name layout. You also want to specify if the name parts will be mandatory, define whether the name parts will render in uppercase, and whether there will be blank lines after the name part.

1. In the Format section on the Create Name Style Format page, click on the **Format Layout** tab.
2. In the Format Variation section, click **Actions** and then click **Add Row**.
3. Enter a variation number in the **Variation** field.
4. In the Format Variation Layout section, click **Actions** and then click **Create**.
5. Complete the fields as shown in this table:

Field	Value
Line	1
Position	1
Prompt	Person First Name
Name Element	First Name
Required	No
Uppercase	No

6. Click **Actions**, then click **Create** to create another address line.
7. Complete the fields as shown in this table:

Field	Value
Line	2
Position	2
Prompt	Person Last Name
Name Element	Last Name
Required	No
Uppercase	No

8. Continue to create as many name parts as you require. Click **Expand** if you want to enter blank lines and delimiters before, or after, a name part. In this section you can also create an Attribute Transform Function to change a data value into a different value.

Assigning a locale

Specify the relevant countries for this name style format.

1. In the Format section on the Create Name Style Format page, click on the **Locale Assignment** tab.
2. Click **Actions** and then click **Create**.
3. Choose United States in the **Country** field.
4. Choose American English in the **Language** field.
5. Click **Save and Close**.

FAQs for Define Data Formats

How can I create an address style format layout?

Click **New** in the Manage Address Formats page and add address lines in the Format Variation Layout section. If you require more than one address style format layout, then you can create a format variation and add a different address format layout.

How can I create a variation of an Address Style Format?

You can create different variations of an address style format, with each variation having its own layout. For example, a country's postal address may have variations such as general postal address, rural address, and military address. When you create an address style format, the layout defined for the format is assigned as the default variation. Click **Add** in the Format Variation region of the Edit Address Style Format page, for the address style format, to add a variation of the Address style. For each address style format variation, you can enter a variation rank to define the priority of the variation.

How can I create a name style format layout?

Click **New** in the Manage Name Formats page and add line entries for each part of the full name in the Format Variation Layout section, within the Create Name Style Format page. If you require more than one name style format layout, then you can create a format variation and add a different name format layout.

How can I create another variation of a Name Style Format?

You can create different variations of a name style format, with each variation having its own layout. For example, different regions in a country may have different name style formats. Click **Add** in the Format Variation region in the Edit Name Style Format page, for the name style format, to add a variation of the name style. For each name style format variation, you can enter a variation rank to define the priority of the variation.

Common Customer Data Management Configuration: Define Contact Points

Telephone Numbering Plan: Explained

A telephone numbering plan defines the expected number and pattern of digits for a country phone number. It defines the country-specific components of phone numbers, display format styles, and codes. This includes country codes, international prefixes, area codes within a country, and mobile prefixes. You can specify country-specific mobile prefixes to determine if a phone number is mobile or not.

You can also create multiple telephone number display formats for a country. For example, for the US phone format, you define the phone country code as 01, fixed area code as three numbers in length, and subscriber number as seven digits in length. You can then create a telephone format for domestic display, such as (999) 999-9999. You can also create a telephone format for international display, such as 999-999-9999.

Updating Telephone Numbering Plans: Explained

You can update telephone number plans from the Edit Telephone Number Plan page. You cannot edit the country information if the country has area codes, mobile prefixes, or phone formats defined.

In the Edit Telephone Number Plan page, you can:

- Edit Scheme
- Edit Regional Information

Editing Scheme

You can edit the country-specific phone information in the Scheme section. You can edit the following fields in the section:

- Trunk Prefix: The code to dial before long distance numbers within the country, for example 1 for the US

- **International Prefix:** The code to dial before international numbers when calling from the country that you are setting up, for example 011 for the US
- **Subscriber Number Length:** The length of the subscriber number for countries where the subscriber length is fixed.
- **Area Code Length:** The length of the area code for countries where the area code length is fixed.

Editing Regional Information

You can edit the regional phone number information for the country in the Regional Information section.

The Regional Information section contains the following tabs:

- **Area Codes:** Allows you to define area codes that you want to use for the country.
- **Mobile Prefixes:** Allows you to define mobile prefixes that you want to use for this country. Mobile prefixes contain the area code and possible prefixes of the subscriber number. For example, the mobile prefix of 650506 consists of the 650 area code and the 506 subscriber number prefix.
- **Telephone Number Formats:** Allows you to define multiple telephone number display formats for a country.

How Setup Telephone Numbering Plans Fits in with the Other Setup Tasks

Telephone Numbering Plans allow you to define the telephone number format for a country. You can have multiple formats for a country, depending on the requirements.

How Telephone Number Plans Setup Fits in with other Setup Tasks

Phone number format, along with the other country-specific location information such as address and name formats, defines geographic-specific data formats. The Manage Telephone Numbering Plans page allows you to create and manage telephone number plans. Oracle Fusion uses the telephone number formats to display phone number in the appropriate style, and to validate phone numbers. You can define country-specific components of phone numbers, display format styles, and area codes. Oracle Fusion uses the phone formats to correctly parse phone numbers and determine the correct format style for displaying phone numbers in user interfaces.

Oracle Fusion validates phone numbers for a country, based on the format defined for the country. The phone numbers are also validated against a common set of rules that apply to all countries. You can also enter time zone information for a country, at the country code and area code level. The time zone information entered with phone number is validated against the time information defined for the country.

FAQs for Define Contact Points

Why can't I update the country information section in the Edit Telephone Numbering page?

You cannot update the country information for a country if the regional information such as area codes, mobile prefixes, and phone formats are defined for the country. For example, if area codes, mobile prefixes, and phone formats are defined for the United States, then you cannot edit the country information.

Common Customer Data Management Configuration: Define Trading Community Import

Trading Community Model Data Import Objects: Explained

Import objects are business entities that can be imported into the trading community model registry, for example, competitors, partners or resource teams. When you create a data import batch you should choose which business entity, or object, you are importing from the batch into the trading community model registry. For example, if you are responsible for resource management, you might want to import objects such as employee resource and resource team.

The import process flow will change according to which object you have selected. There are two import process flows for the following sets of objects:

1. Customer, reference, competitor, and custom party.
2. Employee resource, resource team, partner, and partner contact.

Customer, Reference, Competitor, and Custom Party

When you select these objects you will receive the option to check for duplicates within the import batch before the import, and the option to check for duplicates between the import batch and the trading community model registry before import. You will also be able to choose to preview data before it is imported, specify if addresses will be cleansed before import, and set how many errors you will allow before the import is terminated.

Employee Resource, Resource Team, Partner, and Partner Contact

If you choose to import these objects you will not be able to deduplicate the batch or registry data. However, you will be able to choose to preview data before it is imported, specify if addresses will be cleansed before import, and set how many errors you will allow before the import is terminated.

Using Data Quality Services During Data Import: Explained

Data Quality Services help you ensure the quality of data being imported into the Trading Community Model registry. You can select how to use the services before importing data into the registry.

The data quality services are:

- **Batch Deduplication:** Batch deduplication allows you to define deduplication within the data being loaded.
- **Registry Deduplication:** Registry deduplication allows you to define deduplication of the data being loaded against the records that already exist in the Trading Community Model registry.
- **Import to Registry options:** Import to Registry options allow you to define the import process mode, data cleansing, and geography validation.

You can configure a data import or file import process to invoke these services before importing data into the database.

Defining Data Quality Services for File-Based Import

The file-based import process allows you to define process criteria, file mapping, and import schedule for importing external files containing business objects into staging tables. You can define Registry deduplication options in the file-based import process. You cannot define the other Data Quality Services, such as Batch Deduplication and Import to Registry options, for a file-based import activity during the file-based import process. However, an import activity can be paused and sent for administrator review after preprocessing and before importing the data. The administrator such as a data steward can then review the import process and configure data quality services.

Import activities are paused and sent for administrator review if the **HZ_IMP_PAUSE_FILE_IMPORT** profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page. The administrator can then review and edit import activity details and options. Based on the review, the administrator can decide to approve and schedule the import batch or reject the import batch.

Defining Data Quality Services for Data Import

You can configure data quality services for data import in the Define Import pages. For both the deduplication methods, you can select a match configuration to identify duplicates and also specify the action to perform on the duplicate records. In the Define Import: Import to Registry page, you can specify whether to run the import process in preview mode, whether to cleanse data before import, and whether to perform geography validation of data before import.

You can also configure geography validation for data import at the site level using the **HZ_IMP_DEFAULT_GEO_VALID_ADDRESS** profile option in the Manage Import Profile Options page. The addresses are validated if the profile option is set to Yes in the Manage Import Profile Options page. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported.

Defining the Import Process for Customers and Consumers: Points to Consider

Oracle Fusion Applications provides two utilities to import data, File-Based Import and Data Import. File-based Import allows you to import data directly

from a CSV or XML file. Data Import imports data from interface tables once the tables are populated by an external process. Data Import cannot be used in a Cloud deployment, because you cannot populate interface tables using an external process in a Cloud deployment. In Cloud deployments, you can only use File-based import to import data. However, even if you are using File-based Import, you can use the Data Import dashboard to monitor the import job and view reports and statistics. You can choose to pause a File-Import batch, and use Data Import work area to configure data quality services, monitor import jobs, view errors, and view batch statistics. Import activities are paused if the **HZ_IMP_PAUSE_FILE_IMPORT** profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page.

Before importing data into the registry, you need to decide if you want to use the data quality services and if so, how you want to configure the data quality services.

The data quality services are:

- Batch deduplication
- Registry deduplication
- Import to registry options

Batch Deduplication

Within the batch deduplication page you can decide if you want to identify and resolve duplicates within the batch that you are importing from the interface tables. If you want to check for duplicates you need to choose what match configuration rule you want to use to identify duplicates for each entity. Then you need to specify what action will be taken on the persons, organizations, and address duplicates found within the batch. Your specified actions will be performed on the batch before the data is imported into the registry.

Registry Deduplication

Similar to batch deduplication, registry deduplication identifies duplicates between the data in the batch and the data in the registry before the data is imported into the registry. If you want to check for duplicates you need to choose what match configuration rule you want to use to identify duplicates for each entity. Then you need to specify what action will be taken on the persons, organizations, and address duplicates found in the registry deduplication check. Your specified actions will be performed when you import the batch into the registry.

Import to Registry Options

When defining an import process you can decide whether to run the import process in preview mode, or you can choose to load the data directly into the registry without previewing the data. You can also choose to cleanse addresses prior to import, validate addresses in an import batch against geography data, and define an error limit for the batch.

Import Process Mode

You can choose to run the import batch in preview mode, or you can skip the preview and load the data directly into the registry.

If you select to run the batch in preview mode you will be able to review information about the level of duplicates or incorrect addresses in the batch data before the data is actually imported. You will also be able to preview how many records will be created and how many records will be updated for each entity. You can then continue to import the batch, or you can amend the match configuration rules and actions to be taken on the identified duplicates and then rerun the batch to review the data again.

If you do not want to review the batch data before it is imported into the registry, then you can choose to skip the preview and allow the data to be loaded into the registry as soon as preprocessing is complete. You may prefer not to preview the batch data if the data source is frequently used.

Note

The Define Import: Import to Registry page is the only place that you can specify if you want to run the batch in preview mode. Once the option to skip the preview mode is selected, and you submit the batch for processing, you will not be able to review the batch data before it is imported.

Cleanse Addresses

You can choose to validate the addresses in the interface tables before importing them into the registry. The addresses are validated using an integrated third party service that verifies addresses and corrects them if they are incorrect.

Geography Validation

You can choose to validate the addresses in an import batch against geography data before importing them into registry. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported. However, Geography Name Referencing is run on all imported addresses regardless of this profile option setting.

Error Handling Limit

You can define how many process errors can be generated by the import batch process before the process terminates automatically. Error reports are generated by the application for you to review.

Defining the Import Process for Customers and Consumers: Worked Example

This example demonstrates how to create an import batch, and how to define the import process data quality services you want to use on the data prior to completing the import.

Note

Once an import is completed, the data is loaded into the Trading Community Model registry.

Create an import batch containing customer and consumer objects, load the import data into the interface tables, and configure the data import process so that you can view the batch in preview mode to check that all duplicate data are removed.

Creating an import batch

1. On the Data Import Batches Overview page, click on the **Create Data Import Batch** task located in the task pane.
2. On the Create Data Import Batch page, complete the fields as shown in this table:

Field	Value
Batch Name	Customer Import Batch
Source System	Comma separated values
Object	Customer and Consumer
Estimated Number of Records	300
Batch Description	Import of customer data

3. Click **Save and Close**
4. After creating the import batch, load your data into the interface tables using a standard ETL tool.

Defining the Data Import Process: Batch Deduplication

You want to check the batch for address, organization, and person duplicates; you also want to remove all of these duplicates from the batch.

1. On the Data Import Batches Overview page, highlight the Batch Name, but do not click on the Batch ID URL.
2. Click **Actions** and then click **Import**.
3. On the Define Import: Batch Deduplication page, select the **Check for duplicates within the batch before import**.
4. Complete the fields in the **Select match configuration to determine duplicates within the import batch** section, as shown in this table:

Field	Value
Addresses	Batch Location Basic Duplicate Identification
Organizations	Batch Organization Basic Duplicate Identification
Persons	Batch Person Basic Duplicate Identification

- Complete the fields in the **Override Default Actions** section, as shown in this table:

Field	Value
Select Action for Persons and Organizations	Remove all duplicates
Select Action for Addresses	Remove all duplicates

- Click **Next**.

Defining the Data Import Process: Registry Deduplication

You want to check for address, organization, and person duplicates between the import batch and the Trading Community Model registry; you also do not want to import duplicate records.

- On the Define Import: Registry Deduplication page, select the **Check for duplicates between the import batch and the registry before import**.
- Complete the fields in the **Select match configuration to determine duplicates within the import batch** section, as shown in this table:

Field	Value
Addresses	Batch Location Basic Duplicate Identification
Organizations	Batch Organization Basic Duplicate Identification
Persons	Batch Person Basic Duplicate Identification

- Complete the fields in the **Override Default Actions** section, as shown in this table:

Field	Value
Select Action for Persons and Organizations	Do not import duplicate records
Select Action for Addresses	Do not import duplicate records

- Click **Next**.

Defining the Data Import Process: Import to Registry

You want to configure the data import process so that you can view the batch in preview mode, enabling you to review the data after preprocessing. You want to cleanse addresses before they are imported, and specify an error limit for the batch. You also want to validate all addresses in the import batch against geography data. This validates all incoming addresses as per the geography structure and validation level setup in Manage Geographies task. The addresses that are reported as Error during validation are not imported.

1. On the Define Import: Import to Registry page, select **Run the batch in preview mode**.
2. Select **Cleanse addresses before import**.
3. Select **Validate address against geography**.
4. In the **Error Limit** field, enter 200.
5. Click **Submit**.

Defining the Import Process for Resources and Partners: Points to Consider

Oracle Fusion Applications provides two utilities to import data, File-Based Import and Data Import. File-based Import allows you to import data directly from a CSV or XML file. Data Import imports data from interface tables once the tables are populated by an external process. Data Import cannot be used in a Cloud deployment, because you cannot populate interface tables using an external process in a Cloud deployment. In Cloud deployments, you can only use File-based import to import data. However, you can use File-based Import to leverage Data Import's monitoring and reporting capabilities. You can choose to pause a File-Import batch, and use Data Import work area to configure data quality services, monitor import jobs, view errors, and view batch statistics. Import activities are paused if the **HZ_IMP_PAUSE_FILE_IMPORT** profile option is set to Yes in the Manage Import Profile Options page. If the profile option is set to Yes, all submitted import activities are sent for administrator review and appear in the Data Import Batches Overview page.

When defining an import process you can decide whether to run the import process in preview mode, or you can choose to load the data directly into the registry without previewing the data. You can also choose to cleanse addresses prior to import, validate addresses in an import batch against geography data, and define an error limit for the batch.

Import Process Mode

You can choose to run the import batch in preview mode, or you can skip the preview and load the data directly into the registry.

If you select to run the batch in preview mode you will be able to review information about the level of duplicates or incorrect addresses in the batch data before the data is actually imported. You will also be able to preview how many records will be created and how many records will be updated for each entity. You can then continue to import the batch, or you can amend the match configuration rules and actions to be taken on the identified duplicates and then rerun the batch to review the data again.

If you do not want to review the batch data before it is imported into the registry, then you can choose to skip the preview and allow the data to be loaded into the registry as soon as preprocessing is complete. You may prefer not to preview the batch data if the data source is frequently used.

Note

The Define Import: Import to Registry page is the only place that you can specify if you want to run the batch in preview mode. Once the option to skip the preview mode is selected, and you submit the batch for processing, you will not be able to review the batch data before it is imported.

Cleanse Addresses

You can choose to validate the addresses in the interface tables before importing them into the registry. The addresses are validated using an integrated third party service that verifies addresses and corrects them if they are incorrect.

Geography Validation

You can choose to validate the addresses in an import batch against geography data before importing them into registry. The addresses are validated against the master reference geography data, according to the geography-based address validation settings for each country. The addresses with validation errors are not imported. However, Geography Name Referencing is run on all imported addresses regardless of this profile option setting.

Error Handling Limit

You can define how many process errors can be generated by the import batch process before the process terminates automatically. Error reports are generated by the application for you to review.

Defining the Import Process for Resources and Partners: Worked Example

This example demonstrates how to create an import batch, and how to invoke the data quality address cleansing service you want to use on the data prior to completing the import.

Note

Once an import is completed, the data is loaded into the Trading Community Model registry.

Create an import batch containing partner objects, load the import data into the interface tables, and configure the data import process so that you can view the batch in preview mode to check the address cleansing results.

Creating an import batch

1. On the Data Import Batches Overview page, click on the **Create Data Import Batch** task located in the task pane.
2. On the Create Data Import Batch page, complete the fields as shown in this table:

Field	Value
Batch Name	Customer Import Batch

Source System	Comma separated values
Object	Customer and Consumer
Estimated Number of Records	300
Batch Description	Import of customer data

3. Click **Save and Close**
4. After creating the import batch, load your data into the interface tables using a standard ETL tool.

Defining the Data Import Process: Import to Registry

You want to configure the data import process so that you can view the batch in preview mode, enabling you to review the data after preprocessing. You want to cleanse addresses before they are imported, and specify an error limit for the batch. You also want to validate all addresses in the import batch against geography data. This validates all incoming addresses as per the geography structure and validation level setup in Manage Geographies task. The addresses that are reported as Error during validation are not imported.

1. On the Data Import Batches Overview page, highlight the Batch Name, but do not click on the Batch ID URL.
2. Click **Actions** and then click **Import**.
3. On the Define Import: Import to Registry page, select **Run the batch in preview mode**.
4. Select **Cleanse addresses before import**.
5. Select **Validate address against geography**.
6. In the **Error Limit** field, enter 200.
7. Click **Submit**.

Performing What-If Analysis on Data Import Batches: Worked Example

This example demonstrates how to perform What-If analysis on a data import batch that has been processed and has completed with a status of pre-import completed. The match configuration is redefined and the import process is resubmitted. The batch deduplication actions are then amended, and the batch import is completed.

The following table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
Do you want to redefine batch deduplication match configuration?	Yes, a different match configuration is selected for the organizations entity.
Do you want to redefine registry deduplication match configuration?	Yes, a different match configuration is selected for the persons entity.

<p>What actions do you want to take on Persons, Organizations, and Addresses duplicates?</p>	<ul style="list-style-type: none"> • Within registry deduplication, choose Do not import duplicate records for Persons and Organizations. • Within registry deduplication, choose Import duplicate records for Addresses.
--	---

Prerequisites

1. The data import batch has been created.
2. The data is uploaded into the interface tables.
3. The batch is imported and has completed with a status of pre-import completed.

Viewing the What-If Analysis

1. On the Data Import Batches Overview page, click on the batch ID URL.
2. On the Edit Data Import Batch page, review the summary and import process performance information. Click **Import Details** to open the What-If analysis page.
3. On the Import Process Details page, click the **Batch Deduplication** tab. Check that you are satisfied with the batch deduplication results.
4. On the Import Process Details page, click the **Registry Deduplication** tab. Check that you are satisfied with the registry deduplication results.
5. On the Import Process Details page, click the **Address Cleansing** tab. Check that you are satisfied with the address cleansing results.

Redefining the Match Configuration and Resubmitting the Import Process

The results of the batch and registry deduplication are not as expected and so the match configurations need to be redefined.

1. On the Import Process Details page, click **Cancel**.
2. On the Data Import Batches Overview page, click on the batch name. Click **Actions** and then click **Import**.
3. On the Define Import: Batch Deduplication page, choose a different match configuration for the organizations entity. Click **Next**.
4. On the Define Import: Registry Deduplication page, choose a different match configuration for the persons entity. Click **Next**.
5. On the Define Import: Import Registry page, click **Submit**

Changing the Action for Duplicates within the What-If Analysis

You want to view the What-If analysis for the new match configurations that you selected for the batch.

1. On the Data Import Batches Overview page, click on the batch ID URL.

2. On the Edit Data Import Batch page, review the summary and import process performance information. Click **Import Details** to open the What-If analysis page.
3. The new match configurations have produced satisfactory results, but you would like to change the actions that will be carried out on the duplicates. On the Import Process Details page, click the **Registry Deduplication** tab.
4. For the Persons and Organizations duplicates, choose **Do not import duplicate records** from the choice list.
5. For the possible duplicates for Addresses, choose **Import duplicate records** from the choice list.
6. Click **Complete Import**.

FAQs for Define Trading Community Import

Can I redefine the data import process for an already imported batch and reimport it?

Yes. If the data is still available in the interface tables and the batch status is **Preimport Completed**, **Completed with Errors**, **Error**, or **Terminated when Error Limit Reached**, then you can redefine the data import process and reimport the batch. However, once a batch has been successfully imported then you will not be able to reimport the batch, even if the data is present in the interface tables.

How can I view the errors that occurred during preimport processing?

You can view any errors that occurred after submitting the batch for import by selecting the batch in the data import batches Overview page, and then click **Report**.

What happens if I purge a data import batch?

You permanently remove all records in the batch from the import interface tables. You should purge batches after the batch has been imported successfully and you are sure that the data in the interface tables is no longer required.

Purging the interface tables improves import performance. To archive imported data, you should copy the data to a set of custom tables.

Why can't I purge a data import batch?

Data import batches cannot be purged when they have a status of **Processing**. Purging a batch purges the batch data in the interface tables and so cannot be carried out while the batch is importing the data from the interface tables into the registry.

Why did I receive a message that the data enrichment process was declined?

The Oracle Fusion Trading Community Data Quality service may be unavailable, or you may not have the necessary licenses for the Data Quality service.

Can I try and import a batch again?

Yes. You can reimport a batch to correct validation errors if the import process status is either **Preimport Completed**, **Completed With Errors**, **Error**, or **Terminated When Error Limit Reached**.

Common Customer Data Management Configuration: Define Resource Information

Resource Directory: Explained

The Resource Directory offers detailed information about all the resources within the deploying organization. The Resource Directory also enables you to find and communicate with other resources, and to network and collaborate with them.

Use the Resource Directory to perform the following tasks:

- View and modify your profile
- View your organization and team membership information
- View information related to other organizations and teams
- View the profiles of other resources
- Communicate with other resources

Setting up Resources: Explained

Setting up resources involves identifying a person as a resource and specifying optional profile details as needed. This is an important step because until you identify users as resources, you cannot assign work objects to them.

While identifying a resource is the only mandatory task in resource setup, you may also need to perform some of the following tasks while setting up resources.

- Specify the end date for a resource's engagement with the deploying company
- Assign roles to resources
- Assign resources to organizations
- Assign resources to teams

Identifying Resources: Explained

The Identify Resources step in the Manage Resources task is only needed to identify an existing employee, contingent worker, or partner member as a resource. Usually they are identified as resource in the Manage Users task, or in the Partner Center. If you have created partner members or internal users in the system without making them resources, you can identify them as resources in the Identify Resources step. Until you identify employees, contingent workers, and partner members as resources, you cannot assign them work objects.

Note

Resources need not necessarily belong to an organization, nor do they need to have specific roles assigned. However, it is best to always associate resources with an organization either as managers or as members. Similarly resources should also have at least one role as part of their organization membership. When you identify users as resources, all you indicate is that these new resources can now be assigned work within the deploying company.

Resource Skills and Resource Assignment: Explained

Resource skills help you assign resources to organizations and teams which can best utilize a specific set of skills. For example, if a resource is skilled in a specific technology, product, or business domain, you can assign the resource to teams and organizations that need resources possessing such skills. Use skill-based resource assignment to get the best out of the resources available to the deploying company.

Resources: How They Work within a Team

You can include resources from different resource organizations to work together on a work object as members of the same resource team. You can also include entire resource organizations into a resource team. Generally what resources can do is controlled by their resource organization membership and their hierarchy. Resource teams provide a flexible way of bringing resources together without any organizational or hierarchy-based restrictions.

Assigning Resources to Teams

You can assign identified resources to teams and assign them roles within the team. Each resource can have a specific role within a team. Thus, a resource may play different roles in different teams.

FAQs for Define Resource Information

What happens if I delete a resource?

When you delete a resource, the resource is deleted from the Resource Directory and from the organizations and teams with which the resource was associated. It

is always best, though, to check a resource's usage and remove the resource from all teams and organizations before deletion from the resource directory.

What happens if I add a resource to the organization?

When you add a resource to an organization, the resource becomes a member of the organization. This positions the resource within the organization hierarchy.

Organization membership information is part of the publicly visible details of a resource profile. This means that a resource's organization membership and reporting structure are visible to all active resources within the organization.

If you assign the entire organization to a resource team, all member resources are automatically assigned to the team. This information also becomes part of the resource's publicly visible profile.

What's the difference between a partner resource and an internal resource?

The main difference between an internal resource and a partner resource is the company for whom each works. While the internal resource is an employee or contingent worker of the deploying company, the partner resource is an employee of the partner company.

The methodology used to create resources of these two types is also different. While the partner administrator or channel manager creates a new partner resource through the Oracle Fusion Partner Management applications, internal resources are added using the Manage Users, Hire Employee, or Import Person and Organization task.

Another difference between partner resources and internal resources is that partner resources cannot access the Resource Directory while internal resources can.

Can I create an employee or contingent worker resource?

No. You can only identify existing employees and contingent workers as resources in the Manage Resources task, but you cannot create a new employee or contingent worker in the Manage Resources task.

You can create an employee or contingent worker using Manage Users task, Hire Employee task, or Import Person and Organization task.

What happens when a resource quits?

When resources quit, their status is set to inactive. This automatically removes these resources from all organizations and teams in the deploying company, and no new work objects can be assigned to them.

How can I identify a resource?

To identify specific employees, contingent workers, or partner contacts as resources, you need to search for them and enable them as resources. Once these

new resources have been identified, you can assign them to organizations and teams as needed.

Oracle Fusion CRM or MDM Application Administrators can use the Manage Users setup task to create new employees or contingent worker users. New employees or contingent workers can also be identified as resources similarly.

Common Customer Data Management Configuration: Define Resource Role Information

Define Resource Roles: Explained

Defining resource roles involves defining and configuring the roles that a resource plays as an individual or within a resource organization or resource team in the deploying company. This requires you to specify who a resource is within the enterprise and what specific role the resource performs within the context of an organization or team.

You can assign defined roles to resources directly or to resources within an organization or team context. This action simplifies the task of individually assigning complex roles to resources within the organization.

You can also set several flags while defining roles. Use these flags along with the organization hierarchy information to define the reporting hierarchy of resources. Use the Manager flag to tag a role as a supervisor role. Similarly, attach a Member tag to a role to make it a subordinate role in the hierarchy. Tag roles as Administrator or Lead to indicate the roles that the resource roles have within the hierarchy. Additionally, you can use these flags along with the organization hierarchy information to maintain manager-to-manager relationships within the organization.

Assigning Resource Role Types

Resource role types organize roles into logical groups. This simplifies role assignment and assignment tracking. For example, the Partner resource role type defines a set of partner-specific roles such as partner sales representative and partner sales manager. Use the Partner resource role type to determine the roles that are appropriate for partner members. Similarly, use the Sales resource role type and the Marketing resource role type to categorize the appropriate sales and marketing roles for internal employees or contingent worker resources.

Provisioning Security Roles: Explained

Security role provisioning is the process of automating the provisioning and de-provisioning of security roles based on resource role assignment to resources.

Once security roles are provisioned to resources, they can access the tasks and data enabled for the security role.

Difference between Resource Roles and Security Roles

Resource roles indicate who a person is to the deploying company. As such, resource roles are used for filtering resources and for generating reporting hierarchies in addition to being used to define security policies. A key difference between a security role and resource role is that a resource role may be assigned to a resource without a user account, while a security role can only be provisioned to a resource who has a user account. So while in some cases the resource role may be defined at the same granularity as the security role and used to automate security role provisioning, the resource role concept remains separate from security roles.

Job-to-Role Mapping: Explained

In the Manage Resource Roles task, you can establish job mapping for a resource role. Job-to-resource-role mapping enables you to associate HCM jobs with specific resource roles. This mapping simplifies the task of assigning resource roles to new employees or contingent workers, resulting in time and costs efficiency.

For example, suppose a new employee joins the IT department as a data quality manager. If the new employee's job is already mapped to a resource role like Data Steward Manager, the resource role is automatically assigned when the employee is identified as a resource in the system. This enables you to place new employees faster in organizational and reporting hierarchies. If security roles are also associated with the resource role, then the new employee's access privileges are also granted automatically.

Resource Role Assignment: Explained

Resource role assignment is the task of assigning roles to active resources within the deploying company. These roles are previously set up, and have associated security privileges. Thus, when you assign a specific role to resources, they automatically receive access to specific business functions within the company.

You can assign different roles to the same resource in different organizations. Thus, the same resource can have access to different business functions depending on the security roles associated with the resource roles assigned to them.

Resource Team Membership and Role Assignment Components: How They Work Together

Resources who are team members can be assigned different roles within the team. These roles do not necessarily need to reflect the roles these resources

may play in resource organizations. Depending on the requirements of the tasks assigned to the team, resources may be assigned resource roles, and can accordingly access data related to the tasks in which the team is engaged.

Resource Team Membership

A resource can belong to multiple teams depending on the needs of these teams and the skills that the resource offers. This does not affect the resource's membership with organizations within the deploying company.

Role Assignment

Resources have specific roles to play in every team to which they are assigned. Each of these roles can be different. Also, these roles can be different from the roles assigned to the same resources in resource organizations. Thus, a resource can be a manager in one team, and a member in another simultaneously.

FAQs for Define Resource Role Information

Can I add more than one resource role to a resource?

Yes. While a resource can have only one resource role within a resource organization, a resource can have multiple roles in different organizations across the deploying company.

Can I assign more than one resource role to a resource organization member?

No. One resource can only have one role within an organization. However, a resource can belong to more than one team, and can have multiple roles in each.

Common Customer Data Management Configuration: Define Resource Organization Information

Resource Organizations and Organization Usage: Explained

You can assign organization usage information to resource organizations to classify them based on how they can be used. For instance, resource organizations engaged in sales activities can be assigned the Sales Organization usage. This enables you to sort organizations based on their usage, simplifying your task of working with them.

Primary Resource Organization by Usage Explained

A resource can belong to multiple organizations, and can be assigned different resource roles for each organization. However, a resource can be assigned to one resource organization that becomes the primary organization for the resource as far as a specific usage is concerned.

A resource organization can be said to be a primary resource organization by usage for a resource if the following criteria are met.

- The resource must be a member of the concerned organization.
- The resource organization must be classified as an organization with the specific usage.

For example, a resource organization can be listed as a primary resource organization for marketing for a particular resource if the resource organization is classified as a marketing organization by usage, and has the concerned resource as a member.

Resource Organization Hierarchy Versioning: Explained

As organizations evolve, you may need to make changes to the existing organization hierarchy. Create organization hierarchies to capture these changes without impacting active hierarchies.

Depending on the urgency and nature of the changes within the deploying company, organization hierarchy changes can either be immediate or planned.

Implementing Immediate Resource Organization Hierarchy Changes

In case of immediate changes in the organization hierarchy, either make changes directly to the hierarchy or create a new version of the existing hierarchy and set it to become active when the new organization structure needs takes effect.

Note

Changes made to existing hierarchies are saved automatically and updated immediately.

Implementing Planned Resource Organization Hierarchy Changes

Create a new version of the active hierarchy and specify the date on which the new version needs to become active. Once the new version is saved, you can make and save the changes needed. Ensure that you have made all the changes needed to the new version before the date on which the new version needs to become active.

Organization Memberships and Role Assignments: Explained

When you assign a resource to an organization, you should also specify the role the resource needs to play in the organization. This ensures clarity on the kind of tasks that can be assigned to a resource, apart from defining where the resource is situated within the organization's hierarchy.

Note

There can be only one manager in an organization. However, a resource can be a member of multiple organizations if needed.

Create an Organization with Sales Usage: Worked Example

This example demonstrates how to create a resource organization with sales usage.

you have two choices while creating resource organizations with sales usage:

- Create an organization from an existing department
- Create a new organization

Accessing the Managing Sales and Marketing Organizations Screen

These tasks can be accomplished using the **Manage Sales and Marketing Organizations** screen in the Setup and Maintenance section. To navigate to this screen, perform the following tasks.

1. Click the **Navigator** link at the top-left section of the Oracle Fusion screen.
2. Click the **Setup and Maintenance** link in the Navigator. This displays the Setup and Maintenance screen.

3. If the **Manage Sales and Marketing Organization** task is listed in the **Assigned Implementation tasks**, click the **Go To Task** button against it to access the **Manage Sales and Marketing Organizations** screen. You can also choose to search for the task in the **All Tasks** tab.
- 1.

Creating an Organization from an Existing Department

You need to create an organization with sales usage from an existing department if the concerned department needs to be reused as a sales organization.

1. You can either search for the organization whose usage you wish to modify or click the **Create** button, select the **Create from Existing Department** option and specify the name of the organization in the **Create Organization** screen.
2. Specify the **Business Unit** to which the organization needs to belong and add **Sales Organization** under the Organization Usages section of the screen.

Creating a New Organization with Sales Usage

1. Select the **Create New Organization** option in the **Create Organization** screen and click **Next**.
2. Specify the name and business unit of the new organization and select **Sales Organization** as its usage.
3. Click **Finish** to save the new organization with sales usage.

Adding a Sales Organization to a Sales Organization Hierarchy: Worked Example

This example describes how you can add a sales organization to a sales organization hierarchy.

As new sales organizations are created within your company, you need to add these organizations to your company's sales organization hierarchy. To do so, perform the tasks detailed below.

Decide whether to Create a New Version

1. Navigate to the **View Organization Hierarchies** screen in the Resource Manager.
2. Search for and display the sales organization hierarchy you want to edit.
3. Click the **Actions** drop-down button. If the new organization needs to be added to the sales organization hierarchy with immediate effect, select the **Edit this Hierarchy Version** option. If the new organization needs to be displayed in the sales organization hierarchy only after a specific date, select the **Create Hierarchy Version** option. If you choose the latter option, you need to specify the date from which the new sales organization hierarchy version needs to be active.

Add a New Organization to the Sales Organization Hierarchy

1. In the **Edit Organization Hierarchy Version** page, select the node in the sales organization hierarchy where you need to add the new sales organization and click the **Add** button.
2. Search and select the new sales organization in the **Add Tree Node** popup screen. This adds the new organization to the sales organization hierarchy.
3. Click on the new organization in the hierarchy, drag it to the node where it needs to be placed, and release. The organization is now moved to the location you specified within the selected sales organization hierarchy.
4. Click **Save and Close** to save the changes made to the sales organization hierarchy version and exit the screen.

Assigning a Resource to an Organization: Example

Once you have identified a resource, you can assign the resource to organizations within the deploying company. This example explains how you can assign resources to organizations.

Scenario

You need to assign a resource to an organization. It is important to state here that a resource can belong to multiple organizations at the same time. This means that even if chosen resources already belong to organizations, you can assign them to other organizations as needed.

To assign a resource to an organization, perform the following tasks.

1. Search for the resource using the **Search** controls in the Resource Directory.
2. Display the concerned resource's profile by clicking on the resource's name in the search results.
3. Click the **Organizations** tab in the Resource Profile Details screen. This displays the list of organizations to which the resource belongs.
4. Click the **Create** button to display the **Create Organization Membership** popup screen. Use this screen to search and assign organizations to the concerned resource's profile.
5. Click the **Organization** field and select the organization you need from the drop-down list displayed. You can also enter part of the name of the organization you need and search for it. You can also search for organizations using a range of search options available in the search screen. Click the **OK** button to add the organization you need to the **Create Organization Membership** screen.
6. Click the **Add Row** button in the **Create Organization Memberships** screen. This adds a row to the screen, enabling you to specify the role the resource needs to play in the organization.
7. Click the **Role** field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the

role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.

8. The **From Date** field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.
9. The **To Date** is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.
10. Click the **OK** button to save your changes. This assigns the resource to the organization you chose. You can add the resource to as many other organizations as needed.

You can also assign resources to organizations by searching for organizations and adding members to the concerned organization. To assign resources to organizations by searching for organizations, perform the following operations.

1. Click the **Search** field and, from the drop-down list displayed, select **Organizations**.
2. Search for the organization you need and click on the name of the required organization from the search results. This displays the **Organization** screen, where all the details of the selected organization are displayed.
3. Click the **Members** tab to edit the list of members assigned to the organization.
4. Click the **Create** button to assign a new resource to the organization. This displays the **Add Resource Member** popup screen.
5. Search for the resource by entering the resource's name in the **Resource** field. From the search results displayed, select the resource you wish to assign to the organization and click **OK**. This displays the **Add Resource Member** popup screen with the resource's name added.
6. Click the **Add Row** button to add a role to the newly-added resource.
7. Click the **Role** field and, from the drop-down list displayed, select the role the resource needs to play in the organization. Depending on the role settings created for the selected role, the role type is automatically indicated by a tick mark under the appropriate column.
8. The **From Date** field is automatically set to the current date. You can change this date to reflect the actual assignment date as required.
9. The **To Date** is empty by default. This means that the resource is assigned to the organization you chose infinitely. You can set the date to the actual end date by entering the required value in the mm/dd/yy format.
10. Click the **OK** button to save your changes. This assigns the resource to the organization you chose. You can add as many resources to the organization similarly.

Note

1. The date from which a resource is a member of an organization must be on or after the date on which the resource was hired.
2. If a resource was previously a member of the selected resource organization with an end date later than the current date, and if the

resource was removed from the organization before the end date, you cannot reassign the resource to the same organization.

FAQs for Define Resource Organization Information

Can I create a new organization?

Yes. You can create new organizations using the **Manage Sales and Marketing Organizations** option in the Functional Setup Manager or the Oracle Fusion Partner Management user interface. You can also create new resource organizations from the **User Management** interface. You can also use the interface to manage user accounts and roles, and to create employee and partner-member users.

How can I delete a resource from an organization?

To delete a resource from an organization, you need to delete the concerned resource's membership from the organization.

To manage the membership of an organization, navigate to the **Manage Resource Organization** and select the organization concerned. Once the organization's membership details are listed, you can add, delete or edit them as required.

What happens if I delete a resource from an organization?

When you delete a resource from an organization, you effectively delete the concerned resource's membership with the organization. This also removes the resource from the organization hierarchy.

What's a resource organization hierarchy?

A resource organization displays the relationships between the resource organizations of the deploying company. The resource organization hierarchy is visible to all resources within the company.

Can I create a new organization hierarchy?

No. All organizations within your deploying company are part of a large overarching global hierarchy. Therefore, new organization hierarchies cannot be created. You can, however, move existing organizations or attach new organizations to the existing hierarchy and create a new version of the hierarchy. Once you have created a new version, specify the date from which the new version needs to become active and, on the specified date, your new hierarchy version replaces the older hierarchy.

Can I copy a resource organization hierarchy?

No. You can only have one resource organization hierarchy active at a time within a deploying company. If you need to create a hierarchy for an

organization, you will need to specify the reporting structure within the concerned organization and then make the organization a part of the existing hierarchy.

How can I reuse a resource organization in multiple organization hierarchies at the same time?

There is always one resource organization hierarchy for internal resource organizations and also one partner organization hierarchy for each partner company. You can have one active version and multiple inactive versions of resource organization hierarchies. Resource organizations can be included into any of the active or inactive organization hierarchies.

Can I create multiple resources as managers within a resource organization?

No. An organization can have only one manager.

Can I create a new organization hierarchy version?

Yes. To create a new hierarchy version, navigate to the hierarchy whose new version you wish to create and click on the **Actions** button. From the drop-down list displayed, select **Create Hierarchy Version**. make the changes to the hierarchy as required, specify the dates between which the organization hierarchy needs to be active, and click **Save and Close** to save your changes and exit the screen.

What's a partner organization?

A partner organization represents a department, section, division, business unit, or any other internal or external unit of a partner company registered with the deploying company. Partner organizations provide the framework that enables the deploying company to engage with the partner company while performing sales and marketing activities.

You have to be either the channel manager of the deploying company or a designated partner administrator to create a partner organization. To create a partner organization, you should first register the partner company as a partner of the deploying company. Once registration is complete, you should enroll the partner company into a partner program. Once the partner company has been enrolled into a partner program, you can create and manage partner organizations to model the organization hierarchy of the partner company.

Common Customer Data Management Configuration: Define Resource Team Information

Resource Setup and Resource Role Setup: How They Fit Together

Setting up resource roles helps you define the various roles within your organization so that they can be assigned to resources.

How Resource Role Setup Simplifies Resource Setup

Use the Resource Role Setup tasks to define specific roles that best reflect the way responsibilities are distributed within the deploying company. If your resource roles are created before you start setting up your resources, you can easily assign roles to resources, simplifying task assignment. Resource roles that are tagged as manager or member roles also determine how eventually the reporting hierarchy gets built.

Later, when you need to assign resources to organizations, resource roles can help you specify the precise roles that resources need to play within the organizations to which you assign them.

Resource roles also simplify security provisioning for resources. You can define security roles corresponding to resource roles using the role mapping interface. Once such mapping rules are setup, you can provision security roles by assigning the corresponding resource roles to resources.

FAQs for Define Resource Team Information

What's a resource team?

A resource team is a group of resources formed to work on work objects. A resource team may comprise resource organizations, resources, or both. A resource team cannot be hierarchically structured and is not intended to implement an organization structure. You can also use resource teams as a quick reference to groups of related resources that you can quickly assign work objects to.

Note

Members of teams can either be reassigned separately, or entire teams can be assigned to other tasks as required.

What's a duty role?

A duty role is a set of privileges associated with an aspect of a resource role, and one or more duty roles roll up into a resource role. Basically, resource roles can be said to comprise a set of duties, and each duty has its own set of privileges. Thus, privileges roll up into duty roles, and duty roles roll up into resource roles.

Common Customer Data Management Configuration: Manage Calendar Profile Option

Creating the CRM Common Calendar: Worked Example

The Oracle Fusion Customer Relationship Management (CRM) common calendar is used across CRM applications. The calendar utilizes the Accounting Calendar Default profile option that is **not** set when the delivered product is installed. First, create an accounting calendar with calendar periods appropriate for your CRM needs, and give it a unique name, CRM Calendar, for example. Then, you must specify that calendar in the Accounting Calendar Default profile option. Use the following tasks to set up the CRM calendar.

Creating the Accounting Calendar

1. In the Setup and Maintenance area, Overview page, All Tasks tab, search for the topic names containing **Accounting Calendar**. Manage Accounting Calendars will show up in the results box.
2. Click the **Go To Task** icon.
3. On the Managing Accounting Calendars page, click the **Create** icon.
4. Fill in the required fields as appropriate for the calendar, and click **Next**. Make note of the name utilized because this name will be required in the next topic.

Note

When creating the calendar, the first calendar date must be the first date of the period of the oldest historical data on which you will be reporting. For example, if you select January 1, 2010 as your first calendar date, then you would only be able to enter or import historical data associated with this date and later.

5. Ensure that the calendar data is correct, and click **Save**.

Managing the CRM Common Calendar Profile Option

The CRM calendar profile option must be associated with the new accounting calendar. Follow these steps:

Note

While the Common Financial Calendar feature of Oracle Fusion Applications supports the creation of more than one calendar, Oracle Fusion CRM may only be associated with one calendar. Many features of Fusion CRM utilize this common calendar profile option and changing it could result in the loss of data for one or more applications. Oracle strongly recommends that you do not change the selected Accounting Calendar Default (ZCA_COMMON_CALENDAR) profile option calendar value when it is set.

1. Navigate to the Setup and Maintenance work area, Overview page, All Tasks tab, and search for task names containing **Calendar**.
2. Locate Manage Calendar Profile Option, and click the **Go To Task** icon.
3. On the Manage Calendar Profile Option page, locate the ZCA_COMMON_CALENDAR: Profile Values subpage and click the profile option value line.
4. Click the Profile Value list, and select the calendar name that you created when you created the accounting calendar. Click **Save and Close**.

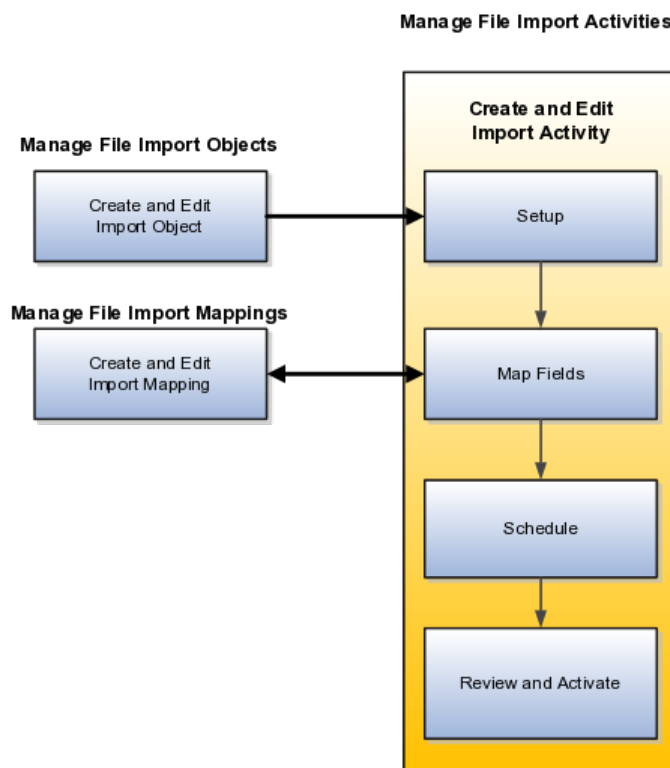
Common Customer Data Management Configuration: Define File-Based Data Import

Define File-Based Data Import

File-Based Data Import Tasks: How They Work Together

The tasks in the **Define File-Based Data Import** task group, available from the **Setup and Maintenance** work area, enable you to import data from text or XML files. This topic explains the role each task plays in the import process.

The following figure provides an overview of the File-Based Data Import group of tasks and their relationships. Each task is explained in the sections that follow.



Manage File Import Activities Task

Use this task to import your data by creating import activities and monitoring their progress. Creating an import activity involves completing the following steps:

1. Set up the import options.
You select the object you are importing, specify the file format and different import options, and upload the file with your data. The options that are available depend on the object you are importing. Some import objects, for example, permit you to upload a ZIP or Java archive (JAR) file of attachments that will be attached to the records in the application after the data file import is complete.
2. Map the fields.
For this task, you create a mapping of the data in your file and the attributes in the application. You can also reuse a mapping you have created in past import activities or use the Manage File Import Mappings task. Any mapping you create in an activity is automatically saved and can be reused in subsequent imports and managed using the Manage File Import Mappings task.
3. Schedule the import.
You can schedule to run the import at a time you specify or run it immediately.
4. Review and activate the import activity.
You activate the import activity by clicking the **Activate** button.

Manage File Import Mappings Task

This task permits you to do the following:

- Manage existing mappings between the data in your import file and the application attributes. For example, you can lock a mapping so that it cannot be modified or edit the mapping to skip attributes during import.
- Create new mappings without a data file.
Creating a mapping using the Manage File Import Mappings task is equivalent to creating it in the import activity, but you must enter any import file column names manually. When you create a mapping as part of the import activity, you upload the file, and the column headings in your file are populated automatically.

Manage File Import Objects Task

This task lets you view all the import objects that have been created for you. All of the objects you can import are already created for you, so you do not have to create more import objects. Creating new objects is reserved for programmers.

Understanding File-Based Data Import: Getting Started

Using the **Define File-Based Data Import** group of tasks available from the **Setup and Maintenance** work area, you can import application data from text or XML files. Your import can be used to create or update records. The data you can import ranges from geography reference data, used to validate addresses in your application, to CRM transaction or legacy data, including customers, accounts, opportunities, and leads. This topic provides an overview of what you must understand to plan your import and prepare your import files. You must read the help topics for each specific import object to learn how to import it.

This topic covers:

- What objects you can import
- Tasks involved in import
- Application business logic associated to the object you are importing
- How data is split into objects and what IDs you must include in your file
- What attributes are available for you to import and their validation

Objects You Can Import

Using file import, you can import a wide range of data listed in the following table. Which of these objects are available to you for selection when you import depends on the applications you have licensed and your security permissions.

Note

To obtain specific information on how to import each of these objects, search help on the term import followed by the object name.

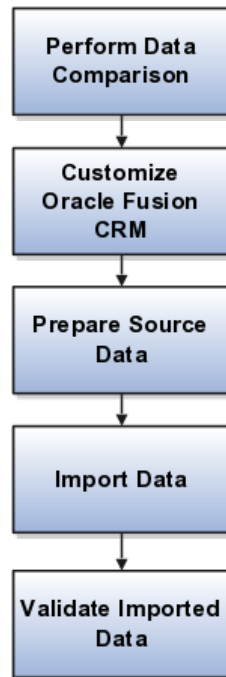
Import Object	Description
Advertising Marketing Activity	Specific instances of an advertising activity for a marketing campaign.
Agreement	Legacy service agreements for use in opportunities and the Oracle Fusion Sales Prediction Engine.
Asset	Legacy customer assets for use in opportunities and the Oracle Fusion Sales Prediction Engine.
Campaign	Marketing campaigns.
Campaign Members	The contacts identified to receive the marketing message for an instance of a marketing campaign activity.
Click-to-Dial Agents	Click-to-Dial agents, including telemarketers, salespersons, and customer service representatives, who use the Click-to-Dial feature.
Click-to-Dial Agents Connector Parameters	Agent connection parameters to access Click-to-Dial functionality.
Consumer	Individuals who are customers or prospects.
Contact	Individuals who are contacts for an existing customer or consumer, or contacts that do not yet have an established association with a customer or a consumer.
Contract	Contracts header information for Oracle Fusion Enterprise Contracts.
Country Structure	Geographical structures for a country.
Customer	Organizations who are sales accounts or prospects.
Customer Hierarchy	Hierarchical structures for customers, such as branches, subsidiaries, and paying relationships.
Customer Hierarchy Member	Identifies the members of the hierarchical structures for customers.
Employee Resource	Use this object to import new users into your CRM organization. You must create users as resources to make it possible for them to participate in CRM business processes.
Event Marketing Activity	Specific instances of a marketing event for a marketing campaign.

Geography	The geographical data for a country.
Group Customer	A group of organizations or persons, such as a household or co-op, with whom you have a selling relationship or a prospective selling relationship.
Incentive Compensation Transactions	Sales compensation data.
Interaction	Records of contacts or communications between internal and external contacts.
Interaction Marketing Activity	Specific instances of an outbound marketing message for a marketing campaign.
Lead	The interest or the potential interest of prospects or existing customers in a product or service being sold.
Legal Entity	Legal entities that have rights and responsibilities under commercial law through the registration with the country's appropriate authority. A legal entity can represent your internal legal structure or dependent authorities such as tax authorities.
Marketing Budget	Business objects that are used to manage funds set aside to fund partner incentives.
Note	Notes associated with objects, such as leads, customers, and opportunities.
Opportunity	Pending sales stored in the application. Opportunities are based on customer interest in a product or service, and can be forecasted.
Partner	Sales partners: independent companies that provide sales and services on behalf of the vendor. In Oracle Fusion Partner Relationship Management, a partner must also be enrolled in a partner program and must be active.
Partner Contact	Contacts for a sales partner.
Product Group	Product group hierarchies that represent the sales catalog for marketing and sales.
Program Enrollments	The enrollment of partners in partner programs, selling programs that provide benefits and incentives for partners.
Promotion	Promotions and coupons which are used to track the effectiveness of marketing campaigns.
Resource Team	Temporary group of resources formed to complete a business task. A resource team can contain both individual resources and resource organizations. A resource team cannot be hierarchically structured and is not intended to implement an organization.
Response	Responses to a marketing campaign.
Sales Account Resource	Employee or partner resources assigned to a customer.
Task	Task that can be assigned to a resource for business objects that support tasks.

Tasks Involved in Import

Importing data involves multiple tasks including the mapping of your data to Oracle Fusion CRM and preparation of the data files to make them ready for import.

The following figure lists the tasks you must perform to import data into Oracle Fusion CRM:



You must:

- Perform a data comparison.
Compare the data you are importing with the attributes and valid values expected by Oracle Fusion CRM.
- Customize Oracle Fusion CRM if required.
If you want to import attributes that do not have an equivalent in Oracle Fusion CRM, then you may be able to add additional fields by using Oracle Fusion CRM Application Composer. For some validated fields, you can change the range of values available for import by editing the lookup types where those values are stored.
- Prepare the source data.
This includes cleaning the data, removing duplicates, and splitting your data into files that correspond to the available import objects. You must ensure that the data meets the specifications required by the attributes in Oracle Fusion CRM, and that the files include the appropriate unique IDs required for import of the object.
- Import the data.
This involves creating import activities by using the Manage File Import Activities task from the **Setup and Maintenance** work area and creating maps between the columns in your files and the attributes in the application.
- Validate the imported data
After the data is imported, you must validate the import by viewing the data in the application.

Understanding Application Business Logic for the Object You Are Importing

You must understand how the data you want to import is used in Oracle Fusion CRM by reviewing the user interfaces, data sheets, and application documentation. If you are importing geography information, for example, you must understand how that geography information is used to validate addresses and what role it plays in sales territories. If you are importing customer data, then you must understand that a sales account in Oracle Fusion applications is a customer with a sell-to address. A customer without a sell-to address is considered a prospect.

Tip

Some of the key business logic pertinent to import is described in the help topics that you can review by searching help or implementation guides on the term import followed by the object name.

Applications help is available as a selection in the **Help** menu on any application page.

Understanding How Data Is Split into Objects and What IDs You Must Include

Before you import, you must also understand how the information you import is represented in Oracle Fusion CRM application objects. Some objects have a simple structure, but others include many child objects, which can be shared with other objects. You must review documentation for each object to understand how to import it.

Importing geography information that forms the basis of address validation, for example, is relatively simple because the relationships between the geographical data is determined within the data itself and the number of attributes that determine an address is small.

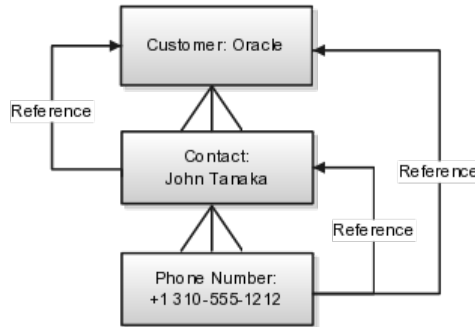
Importing customer data is much more complex, requiring you to understand how customer information is broken down into different objects and how those objects must reference each other in your file.

Some legacy applications store information about customers and competitors as separate entities. If a competitor becomes a customer, you have to copy the information about the organization and create a customer record. In Oracle Fusion, you indicate a competitor is now a customer. Similarly, contact information is often part of the customer information so you must reenter the information about that person each time he changes employers. In Oracle Fusion, every individual and organization is a party in the model. If a contact moves to work for a different customer, you simply indicate that the contact works for someone else. You get to keep all the information you have about that contact. This means that customer data is split into many objects that all have to reference each other.

For example, customer contacts with phone numbers are made up of a hierarchy of three objects: the customer, the contact, and the phone number. Each object must have a unique ID and has to refer to the objects above it in the hierarchy when you import.

The following figure provides an illustration with a simple example.

- Oracle is the customer at the top of the hierarchy with its unique organization ID.
- The Oracle contact, John Tanaka, has its own unique ID and must also reference the organization ID for Oracle.
- The contact's phone number, +1310-555-1212, must have its own ID and reference both the contact and the customer.



Splitting data into multiple objects impacts the content of the import file:

- Your file must include many columns of IDs
If your legacy data does not already contain the required IDs, you can create them manually in Microsoft Excel.
- Each row of your file must include the unique ID of the object you are importing and the IDs of all the parent objects.
- You must import data in a specific order to make sure that a parent object exists before you import the child object or you must include the parent object information in the same file.

For example, if you want to import customer contacts, then you must ensure that all the customer records already exist or that you include the customer data in the same file.

Having multiple reusable objects also affects the way you create the mapping between your file and the application attributes:

- When you create an import activity and map the attributes in your file to the attributes in the application, you must select the child objects you are importing before selecting the attributes.
- You will need to review separate documentation for each object to find out which attributes are available for import and what values are accepted.

Understanding What Attributes Are Available for Import and Their Validation

You must also understand what attributes you can import for each object, which attributes are required and must be present in your file, and what attributes values are permitted.

The list of attributes, their descriptions, and information about valid values are provided in application help topics by import object and in documentation available in the Oracle Enterprise Repository.

Note

The valid values that must be present in your import file are not necessarily the same as those you see in the application. For example, values stored in lookup types, require you to include the lookup code for each value rather than the meaning displayed in the user interface.

To obtain the list of valid import values for validated fields, you must consult the lookup types where those values are set. If the lookup type is extensible, then you can edit it to match the values you are importing.

Some import objects permit you to add additional attributes using Oracle Fusion CRM Application Composer. For these objects, you first create the additional attributes in the composer and then create your mapping.

Importing Data from a File: Getting Started

Using the **Define File-Based Data Import** group of tasks available from the **Setup and Maintenance** work area, you can import a wide range of application data from text or XML files. Your import can create or update application records. Use the steps outlined in this topic as a general guide only. Import options and other details differ by the type of data you are importing. Before you import, you must also understand how the data in your file will map to the attributes in the application and what values are expected as described in the related topics.

To help you get started, you can use the example templates provided in Document ID 1503223.1 on My Oracle Support. The templates include columns for the most common application attributes and reference numbers. To simplify mapping of source data to the target table columns, each template column is labeled with the combination of the target object and attribute you must select.

General Steps for Importing Data from a File

To import data from a file:

1. In the **Navigator** menu, click the **Setup and Maintenance** link located under the **Tools** heading.
The Setup and Maintenance work area appears.
2. From the **All Tasks** tab in the **Overview** region, search for the Manage File Import Activities task.
3. Click the **Go to Task** button for the task.
The Manage Import Activities page appears.

Tip

You can also navigate to the Manage Import Activities page by selecting the import task in the work areas for objects that support importing data.

-
4. Click the **Create** button.
The Create Import Activity: Enter Import Options page appears.
 5. In the **Name** field, enter a name for your import.
 6. In the **Object** field, select the object you are importing.
 7. In the **Source File** region, select one of the **Upload From** options.

Note

If you are importing a file into one of the Oracle Cloud services, then you must upload files from your desktop.

-
- **Desktop**
After you select this option, you must browse for the file.
 - **URL**
 - **Network**

Enter the path with the following preface file://

For example, if the file is located at /tmp/sample.csv , then your entry is file:///tmp/sample.csv. Note the extra slash in the entry.

8. If your data file includes a header row, then select the **Header Row Included** option.

While you can upload files without header rows, doing so makes it more difficult to complete the mapping between the data in your file and the application.

9. If your file does not use a comma to separate values, then select the correct delimiter in the **Data Type** field.
10. The import requires you to create a mapping between the data in the import file and the target attributes in the application. If you previously imported a file with the same format as the file you are importing now, then select an existing mapping from the **Import Mapping** list. By default, the application saves the import mapping under the import activity name appended with the activation date and time. If you are importing a file in this format for the first time, then you must create the mapping in the next step.
11. If you are importing one of the following objects, you can import attachments by selecting the **Browse** option in the **Attachments** region and selecting a Zip or JAR file of the attachments.

The following import objects support the import of a single attachment for each record:

- Assets
- Interactions
- Opportunities
- Partners

The following import objects support multiple attachments for each record:

- Leads
- Tasks

Note

You must include a column with the names of individual attachments in your import file as described in the Importing Attachments with File-Based Data Import: Explained topic.

-
12. In the Import Options region, some of the options in the following table may be available depending on the object you are importing

Option	Description
Import Mode	<p>For importing leads and employee resources only, you have the option of specifying if you want to create and update records or update only.</p> <p>If you select update, then any new records will be ignored by the import process.</p> <p>For all other import objects, both create and update are available.</p>
Allowable Error Count	The maximum number of errors before you terminate import. Below the threshold, all records without errors are imported. Above the threshold, no records are imported.
Notification E-Mail	The e-mail address of the individual who will receive import processing notifications.
Customer Data Management Duplicates	This option is available only if you have licensed Oracle Fusion Trading Community Data Quality. You can select different options for handling duplicates when you are importing customers, consumers, and legal entities either alone or as part of another object.
Decimal Separator	Decimal separator used in your import file.
Date Format	The formatting of the date fields in your file.
Time Stamp Format	The formatting of time fields in your file.
File Encoding	The overall encoding for your file.

13. Click **Next**.

The Create Import Activity: Map Fields page displays the attributes in your file with some sample data. If you are reusing an import mapping, then both the **Source** and **Target** columns are already populated. The **Source** columns represent your file. If you are creating a new mapping, then the **Target Object** and **Attribute** columns are blank.

14. For each column in your file that you want to import, select the target object and one of its attributes. Each import object comes with its own set of attributes.

You can choose not to import a column in your file by selecting the **Ignore** check box.

15. If you want to specify a constant value for an attribute in the application, you can add the object, the attribute, and its value in the **Set Constant Values** region.

Note

If you are importing data using the import templates provided on My Oracle Support, then each column header in your template includes the names of both the target object and the target attribute to simplify your mapping process.

16. Click **Next**

The Create Import Activity: Schedule page appears.

17. By default, the import will start immediately after you click **Activate** in step 19. You can instead start the import at a time you specify by selecting **Specific Date** from the **Schedule** list and entering a start date.

18. Click **Next**.

The Create Import Activity: Review and Activate page appears.

19. Click **Activate**.

You return to the Manage Import Activities page where you can view the status of your import.

An import activity with the status of **Completed** indicates that your data was loaded into the application tables.

20. If the process completes with the status of **Completed with Errors**, click the status link.

You can view the errors at the bottom of the View Import Status page and by selecting the Attachments link in the Files Processed region.

Importing Attachments Using File-Based Data Import: Explained

When you import data from a file using the **Manage File Import Activities** task, you can include attachments for a subset of import objects, including leads, opportunities, tasks, and other objects. This topic lists the objects that support attachment import, explains how the import works, and provides guidelines.

Objects That Support Importing Attachments

A subset of import objects supports importing attachments. Some objects permit you to import only one attachment for each record. A few objects permit the import of multiple attachments.

The following table lists the import objects that support the import of attachments. A subset of these supports the import of multiple attachments for each record, as indicated in the following table.

Import Object	Single or Multiple Attachments for Each Record
Assets	Single

Interactions	Single
Leads	Multiple
Opportunities	Single
Partners	Single
Tasks	Multiple

How You Import Attachments

To import attachment, you:

1. Create a ZIP or JAR file of the attachment files.

Note

The attachments can be organized in nested folders.

2. Add an additional column to the data file you are importing, and enter the names of the attachment files. The header of this column must always be: ATTACHMENT_FILE_NAME.

This reference column specifies which attachment file belongs to which record. Other optional columns listed in the following table can be used to upload other attachment information, such as a brief description.

The following table lists all of the available attributes with the header names you must use.

Column Header	Required or Optional	Description
ATTACHMENT_FILE_NAME	Required	Captures the file names in the ZIP file.
ATTACHMENT_FILE_DESC	Optional	Short description of the file.
ATTACHMENT_FILE_TITLE	Optional	File title. If you do not import a file title, then the application uses the file name as the title.
ATTACHMENT_CATEGORY_NAME	Optional	Attachment category.

3. Upload the ZIP or JAR file with the attachments when you create the file import activity.

You do not map the attachment attributes. The application identifies the special names in the column headers.

Note

You must upload one ZIP file of all the attachments. You cannot upload attachments in multiple files or add additional attachments after you upload the ZIP file the first time.

Importing Single Attachments

The following example illustrates the steps required for importing one attachment for each record.

1. Create a ZIP or Java archive (JAR) file of all of your attachments, for example: myfiles.zip:
 - file_1.doc
 - file_2.doc
 - file_3.doc
 - file_4.doc

Note

The numbers in the file names are provided for this example. They are not required.

2. In the import file, add the additional columns to capture the attachment information.

At a minimum, you must include a column with the header name: **ATTACHMENT_FILE_NAME** to capture the attachment file names. You must use this header name without variation so that the application will identify it.

Unlike the other attributes in your import file, you do not map the attachment columns to import objects and attributes when you create an import activity. The application does the mapping automatically. The mapping is based on the column name.

Your import file might contain data similar to that shown in the following table.

LeadName	StatusCD	Score	ATTACHMENT_FILE_NAME
Lead 1	QUALIFIED	7	file_1.doc
Lead 2	QUALIFIED	8	file_2.doc
Lead 3	QUALIFIED	3	file_3.doc
Lead 4	QUALIFIED	7	file_4.doc

3. When you create the import activity to import your file, you upload the ZIP or JAR file from your desktop by clicking the **Browse** button in the Attachments region in the Import Options page.

The ZIP or JAR file is imported at the same time as the data file.

Note the following:

- If a record does not have an attachment, then leave the column value blank.

- You can specify a single file to be the attachment for more than one or for all records you are importing. For example, if you wanted the same file to be attached to all the four leads, then your import file structure would be like the data in the following table.

LeadName	StatusCD	Score	ATTACHMENT_FILE_NAME
Lead 1	QUALIFIED	7	file_1.doc
Lead 2	QUALIFIED	8	file_1.doc
Lead 3	QUALIFIED	3	file_1.doc
Lead 4	QUALIFIED	7	file_1.doc

Importing Multiple Attachments

For import objects that support multiple attachments, you repeat the column header names.

Note

If you are adding any of the optional attachment columns, then you must repeat all the columns in the same order.

Suppose you are importing four leads:

- Lead 1 includes attachments file_1.doc, file_2.doc, file_3.doc.
- Lead 2 includes the attachment file_4.doc.
- Lead 3 includes attachments file_5.doc, file_6.doc, file_7.doc.
- Lead 4 has no attachments.

Your data file might look like the data in the following table.

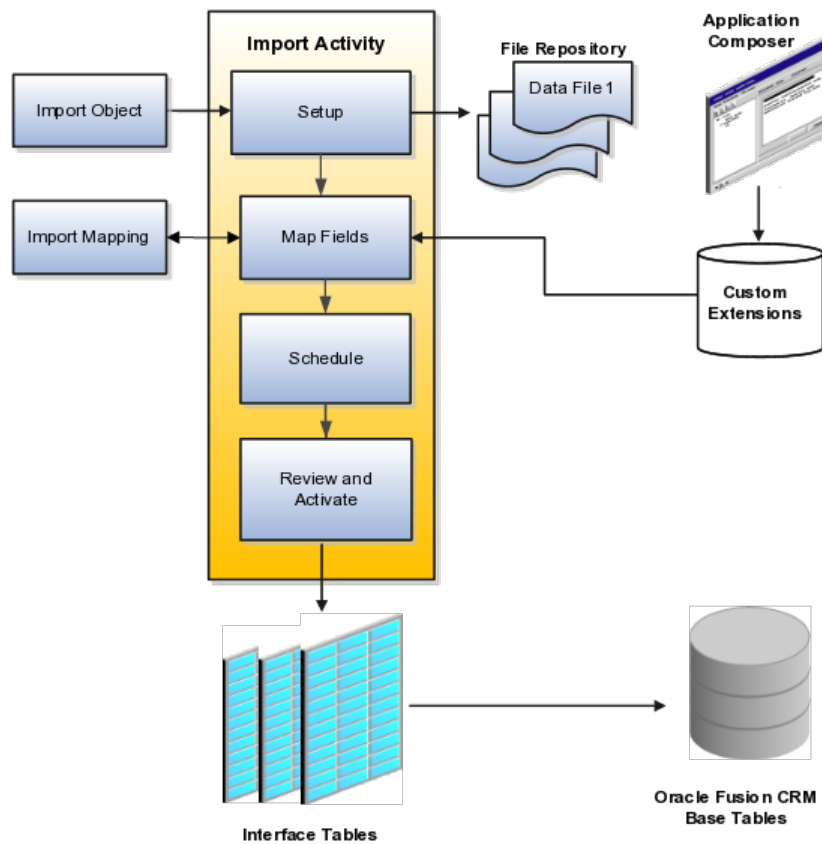
LeadName	StatusCD	ATTACHMENT_FILE_NAME	ATTACHMENT_FILE_NAME	ATTACHMENT_FILE_NAME
Lead 1	QUALIFIED	file_1.doc	file_2.doc	file_3.doc
Lead 2	QUALIFIED	file_4.doc		
Lead 3	QUALIFIED	file_5.doc	file_6.doc	file_7.doc
Lead 4	QUALIFIED			

File-Based Data Import: How It Works

The Define File-Based Data Import group of tasks relies on integration with different Oracle Fusion Applications and architecture components, such as interface tables and application base tables. This topic provides an overview of these components to help you understand the import process and the different import activity statuses.

The following figure provides an overview of the major application components used when you import data from a file. These components include

- Import object
- Import mapping
- Import activity
- File repository
- Oracle Fusion CRM Application Composer
- Interface tables
- Application tables



Import Objects, Import Mapping, and Import Activity

The import objects you select when you create an import activity are provided by Oracle. They are managed using the Manage File Import Objects task.

When you create an import activity, you must specify a mapping of the fields in your file to the attributes of the import object. You can create the mapping either while creating an import activity or separately using the Manage File Import Mappings task. The mapping is stored and managed as a separate object.

File Repository

The text or XML data file you upload for import is stored in a file repository so that it is available for import processing when you schedule an import activity. Any attachments you upload are stored in the same repository.

Application Composer and Customer Extensions

When you create additional attributes for import using Oracle Fusion CRM Application Composer, these extensions are stored in a separate repository and are available when you create your mapping.

Interface Tables

The import activity populates the application interface tables with your data. This is an intermediate holding area for your data that permits you to correct any import errors. Oracle Fusion CRM application data is not changed at this point.

Application Tables

The import activity loads your data into Oracle Fusion CRM application tables to complete the import.

File Import Activity Statuses: Explained

This topic explains the meaning of the different import activity statuses when you import data from a file using the Manage File Import Activities task.

The following table lists and describes the import activity statuses.

Status	Description
No status displayed for the activity	The new import activity definition is not complete.
New	The import activity definition is complete, but the activity is not yet scheduled to run.
Scheduled	The import activity definition is complete and scheduled to run later.
Interface table upload in progress	The data file is being loaded into the interface tables.
Base table upload in progress	The data file is being loaded from the interface tables into the application tables.
Completed	All records were processed and loaded into the application tables.
Completed with errors	The import activity found errors in the data file that you must correct. You can view details about the errors by clicking the status link.

File-Based Import Objects: Explained

Import objects represent the application and attribute information for business objects that can be imported using external source files.

This topic describes the following:

- Import object management options
- Custom objects

Import Object Management Options

A single import object can have multiple associated components that are considered objects by themselves. An object and associated objects that can be imported within the same source file are grouped together within the application module class.

Note

Each object includes the Import Activity object (MktImpJobs1). The Import Activity object is a required component of the application module but is not mapped to a source file. All values for this object are derived from the Import Activity definition. Consequently, do not update the **Map**, **Required**, and **Default Value** settings for the Import Activity object.

The following table includes information about the import object:

Option	Description
Attributes	A view-only listing of object attributes that represent each column in the interface table for the object.
Length	A view-only listing of widths for the columns in the interface tables. If the source file values for the attribute have more characters than the attribute length, the source file row will not be imported.
Default Value	Optionally, specify an attribute value to use if a value is not available from the source file or Import Activity constant value.
Map	Enable the list of attributes that can be mapped to a source file or constant value in the Import Mapping and Import Activity Map Fields step.
Required	<p>Specify the list of attributes that must be mapped to source file columns. Consequently, if you have selected an attribute as required, you must also enable the Map option for that attribute.</p> <p>When mapping the external source file, the required target attribute defined for the object are displayed with an asterisk.</p>

Custom Objects

To use the file-based import feature for custom objects, you must first generate the artifacts required for import. You generate these required artifacts within Oracle Fusion CRM Application Composer, after making your object model extensions.

File-Based Import Mapping: Explained

Import mapping enables you to predefine a mapping between the columns provided in a source file and the attributes pertaining to the objects being imported. Once you create a mapping, it can be reused in the Import Activity definition.

This topic contains the following sections:

- Import options
- Source file options
- Target options

Import Options

The following attributes pertain to the import mapping.

Attribute	Description
Object	The business object to be imported.
Name	The name that identifies the mapping in the Import Mapping and Import Activity UIs. If the mapping was initially created while mapping fields directly in the Import Activity user interface and automatically saved without providing a user-defined mapping name, the mapping name is derived from the Import Activity name and date.
Decimal Separator	The format of the fractional portion of numerical values in columns mapped to attributes with a decimal attribute type.
Date Format	The format of values in columns mapped to attributes with a date attribute type.
Timestamp Format	The format of values in columns mapped to attributes with a time stamp attribute type.
Lock	If selected, prevents any user, other than the creator of the mapping, from editing the mapping.

Source File Options

Map each column that the source file is expected to contain with a specific attribute.

The following table describes the details pertaining to columns provided in the source file:

Source Column	Description
Sequence	The sequence number in which the columns are expected to be provided in the source file. Two rows cannot have the same sequence number.
Column Name	The column name expected in the source file if a header row is included, or more generic values such as Column A, Column B, and so on, if the header row is not included for Text file types. The tagging structure is represented for XML file types.
Column Width	Use when the delimiter value is fixed width for Text file types only.
Ignore	Ignore the source file column to exclude the data from being imported.

Required	If selected, a value must exist in the source file or the row will not be imported.
----------	---

Target Options

The following table describes the details pertaining to corresponding attributes in the target application table:

Target Attributes	Description
Object	The group of import objects that represent the components of the business object being imported.
Attribute	The attribute name that represents the corresponding interface table column for the object.
Duplicate Validation	If selected, the attribute, along with other selected attributes, determines what constitutes a duplicate object when comparing objects in the interface tables and existing objects in the target application tables. For example, to validate the uniqueness of an object in the target application tables by the combination of an object's name and date, select Duplicate Validation for both attributes in the mapping.

File-Based Import Processing: How it Works

The file-based data import process includes processing the source file data and inserting it into the interface tables, moving the interface table data into the destination application tables, and then processing the attachments for the imported objects. Processing factors are subject to the settings defined for the Import Activity, Mapping, and Import Object. You can monitor the processing steps and view process reports for each Import Activity.

This topic describes the following:

- Inserting Data in the Interface Tables
- Interface Table Data Validation and Error Counts
- Interface Table to Destination Application Table Processing
- Importing Attachments
- Viewing Import Results

Inserting Data in the Interface Tables

Data exists in various sources and in various formats. The file import processing starts with reading the source data, parsing the data, and inserting into the appropriate interface tables. The source of the data comes from the following:

- Source file values mapped to target object attributes in the Import Activity.
- Constant values defined for target object attributes in the Import Activity.
- Default values defined for target object attributes in the Import Object.

Interface Table Data Validation and Error Counts

The data is initially validated against the predefined Import Mapping and the Import Object settings as the interface tables are being populated by the initial file import process. The interface table data is validated again before importing into the destination application tables.

- Validation includes:
 - Missing required values
 - Values that exceed the attribute length
 - Invalid values
 - Duplicates to existing records in the destination application tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping.

Note

For the Lead import object, the duplicate checking is only done for existing leads created within the look back days setting of the Import Activity.

-
- Duplicates to existing records in the destination application tables for Customer Data Management objects based on Matching Configurations.
 - Errors

Most validation issues are recorded as errors, with the exception of Customer Data Management duplicates found during the Matching Configuration process. In this case, matched records are only considered as errors if:

 - Customer Management Duplicates option is set to **Do Not Import** for the Import Activity and
 - The main object of the Import Activity is a consumer, customer, or legal entity object
 - Allowable Error Count Threshold

The validation of the interface table occurs before any records are imported into the destination application tables. Once the validation process has completed, the count of records with errors is compared to the Allowable Error Count Threshold value specified for the Import Activity. A count above the threshold will stop the import process for all records. If the count is below the threshold, records without errors will import. In either case, records with errors will be reported in the Error and Exception files.

Interface Table to Destination Application Table Processing

The import process orchestrates the import for each of the component objects that make up the overall main objects of the Import Activity.

Importing Attachments

Once the objects have imported successfully, the attachments are processed. The import process matches the source file attachment name to the file name included in the compressed file entered on the Import Activity. The attachment file is imported into Universal Content Manager and then associated as an attachment to the imported object.

Viewing Import Results

You can monitor all file-based Import Activities that are currently scheduled to run, have completed successfully, or failed with errors. For each Import Activity, you can view the details pertaining to each underlying process. Once an Import Activity process has completed, the following processing reports are added as attachments to the process:

- Log file. Includes the records that were successfully imported plus the unique destination application table identifiers for the objects.
- Exception file. Includes the records that were not imported plus a reference to an error for each record that failed validation.
- Error file. Includes all the errors for each record that failed validation.

Import Activity Source File Options: Explained

The Import Activity consists of a step by step guided process to assist you with creating an import activity for a given object.

This topic describes the source file options defined in the Import Activity that are used by the import process to locate and parse the source file data.

Source File Data

Enter attribute details pertaining to the source file as follows:

Option	Description
File Type	Source file must be either Text or XML.
Data Type, Delimiter, and Header Row Included	A Text file type can further be defined based on how the data is delimited and if the source file is expected to include a row of headings for each column.
Import Mapping	Displays a list of predefined mappings for the object selected for this import activity. The selected mapping will be used as the basis for mapping your source file in the next Import Activity step.

Source File Location

The following outlines the options that are available to you when locating your source file for import.

Option	Description
File Selection	<p>Select from the following file selections:</p> <ul style="list-style-type: none"> • Specific file <p>Enables you to upload a specific source file from a local file system, such as your desktop, a URL address, or from a network path. A file name is required for this option.</p> • Most recent file <p>Enables you to schedule repeating import activities without having to select a new file every time. This selection is only available when you select Network from the Upload From options.</p> <p>You need to copy the new file to the specified network path for repeating import activities. You do not need to enter a file name for this option and can only upload your source file from a network path. The asterisk wildcard is supported for multiple characters. The question mark wildcard is supported for a single character.</p>
Upload From	<p>You can upload the source file from three locations:</p> <ul style="list-style-type: none"> • Desktop • URL • Network <p>If you select Desktop, a File Name field with an associated Update button is displayed. Click Update and browse to search for and select the file you want to upload.</p> <p>If you select URL, enter the address location as in the following example format: <code>http://www.example.com/</code></p> <p>If you select Network, enter the file name path as in the following example format: <code>\ComputerName\SharedFolder\Resource\</code></p> <hr/> <p>Note</p> <p>If you selected the Specific File as your file selection option, then you will have to include the file name for both URL and Network file path locations.</p>

Import Activity Import Options: Explained

This topic describes the import options available while creating import activities.

Source File Data Transformation

The following options are used to identify the formatting of source file data so the data can be correctly interpreted and transformed by the import process:

Option	Description
Decimal Separator	The format of the fractional portion of numerical values in columns mapped to attributes with a decimal attribute type.
Date Format	The format for values in columns mapped to attributes with a date attribute type.
Time Stamp Format	The format for values in columns mapped to attributes with a time stamp attribute type.
File Encoding	The overall encoding of the characters within the file.

Interface to Target Import Options

Depending on the object you are importing and the application modules you have implemented, you can select one or more of these options.

Option	Description
Import Mode	<p>For importing leads and employee resources only, you have the option of specifying if you want to create and update records or update only.</p> <p>If you select update, then any new records will be ignored by the import process.</p> <p>For all other import objects, both create and update are available.</p>
Allowable Error Count	<p>An error count above the threshold will stop the import process for all records. If the error count is below the threshold, records without errors are imported. In either case, records with errors will be reported in the Error and Exception files.</p> <p>Validation errors include:</p> <ul style="list-style-type: none">• Missing required values• Values that exceed the attribute length• Invalid identifiers and lookup codes• Duplicates to existing records in the destination tables based on the combination of attributes selected for duplicate validation in the predefined Import Mapping <p>Duplicates found using matching configurations for Customer Data Management objects do not contribute to the error count.</p>
Notification E-Mail	The e-mail of the intended recipient of import processing notifications.

Customer Data Management Duplicates	<p>Using this option, which is available only if you have licensed Oracle Fusion Trading Community Data Quality, you can select different options for handling duplicates when you are importing customers, consumers, and legal entities either alone or as part of another object.</p> <p>The duplicates are determined using the following matching configurations:</p> <ul style="list-style-type: none"> • Batch Location Basic Duplicate Identification • Batch Person Basic Duplicate Identification • Batch Organization Basic Duplicate Identification <p>You can select from one of the following:</p> <ul style="list-style-type: none"> • Do Not Import Duplicate Records <p>If the main object of the Import Activity is a consumer, customer, or a legal entity object, rows that are matched to existing records will not be imported. These duplicate records are reported in the Exception and Error reports.</p> <p>If the Customer Data Management objects are components of another object and one or more matches are found, the existing duplicate records are evaluated to determine the most recent record. The most recent record will be associated with the main object being imported.</p> <p>For example, when importing a marketing response object, the consumer object is also a component of the response. If the consumer is matched to an existing record, the consumer in the interface tables is not imported. However, the response object will import and the most recent existing consumer record will be associated to the response.</p> <ul style="list-style-type: none"> • Import Duplicate Records <p>The Customer Data Management objects will be imported even if matched records exist.</p> <ul style="list-style-type: none"> • Import Duplicate Records and Create Resolution Request <p>The Customer Data Management objects will be imported even if matched records exist. In addition, a duplicate resolution request is created and displayed in the Customer Data Management, Duplicate Resolution work area.</p>
-------------------------------------	---

Duplicate Look Back Days	This option applies only to the Lead import object. Only existing leads created within the period determined by the look back days value are evaluated for duplicates based on the attributes selected for duplicate validation in the predefined import mapping. If a duplicate is found, the lead will not be imported and the duplicate record will be reported on the Exception report. Duplicate leads are included in the calculation of the allowable error count threshold.
--------------------------	---

Import Activity Field Mapping: Explained

After entering your import options, the second step of the import activity process is to map fields in the source file to the corresponding target attributes.

This topic explains:

- Map Fields
- Saving the Import Mapping
- Constant Values

Map Fields

The Map Fields section can be subdivided into source file columns and target attribute columns.

The source column header value is derived from one of the following:

- Predefined mapping, if one is selected
- The source file, if the **Header Row Included** option is selected in the first step of the Import Activity definition (for Text file type only)
- Generic values of Column A, Column B, and so on, if the **Header Row Included** option is not selected (for Text file type only)
- XML tagging structure (for XML file type only)

The following table outlines the source columns:

Source Column	Description
Column Header	Represents the column header for Text file types and the tagging structure for XML file types.
Example Value	Values are derived from the first source file saved with the predefined mapping. If you did not select a predefined mapping, the example values are taken from the first data row in the source file selected in the first step of the Import Activity definition.
Ignore	Select this option if you do not want to import the source file data in that column.

The following table outlines the target columns:

Target Column	Description
Object	The group of import objects that represent the components of the business object being imported.

Attribute	The attribute name that represents the corresponding interface table column for the object.
-----------	---

Saving the Import Mapping

The mapping between source file information and target attributes is saved as a reusable mapping when the Import Activity is saved, using the import activity name and date to derive a mapping name. If you selected a predefined mapping, modifications made in the Import Activity to an unlocked mapping will update and save to the predefined mapping. If the predefined mapping is locked, a modified mapping will be saved as a new mapping. To specify a mapping name for new mappings, select the **Save As** option from the Map Fields **Actions** menu.

Constant Values

Constant values provide a way to specify a value for a target attribute that all imported objects will inherit. For example, if a source file does not contain a column for business unit and all of the objects in the file belong to the same business unit, enter a constant value for the object and business unit attribute.

File-Based Import Monitoring: Explained

You can monitor all file import activities that are currently scheduled to run, have completed successfully, or failed with errors. For each import activity, you can view the details pertaining to each underlying process and make necessary updates for any failed records to import again.

You can view the list of import activities from the Manage Import Activities page. Select the import activity that you want to monitor by clicking on the hyperlink in the corresponding Status column. The View Import Status results page is displayed which contains the following sections:

- Files Processed
- Import Processes

Files Processed

The Files Processed section displays a row for each source file that is processed.

The import processing details are summarized and displayed for each source file and include the following:

File Processing Summary Information	Description
Records Read From File	The number of records read from the source file.
Format Errors	The number of errors found when processing data to insert into the interface tables from the source file, Import Activity constants, and Import Object value default values. View the error details in the Exception and Error files attached to the process.
Load Errors	The number of errors found when importing data from the interface tables to the destination application tables. View the error details in the Exception and Error files attached to the process.

Successfully Loaded	The number of import objects imported to the application destination tables. If the import object is made up of multiple components, each component is counted as successfully loaded. Consequently the Successfully Loaded count may be larger than the Records Read From File count. View the successful record details in the Log file attached to the process.
Attachments	Once an Import Activity process has completed, processing reports are included in the Attachments column. The Log file includes the records that were successfully imported plus the unique destination application table identifiers for the objects. The Exception file includes the records that were not imported plus a reference to one of the errors for each record that failed. The Error file includes all the errors for each record that failed validation.

Import Processes

From the Import Processes section, you can view details pertaining to each process involved in importing the objects in the source file. A listing of brief messages provides information on processing steps within each underlying process.

Getting Started with File-Based Import: Documentation Overview

This topic provides an overview of the types of help available for importing legacy and other data into Oracle Fusion CRM using File-Based Data Import.

There are three main sources of information for import:

- Help topics about using the File-Based Data Import tool are linked to the application pages and available by clicking the help icon.
These topics provide general information about import.
- Help topics on individual import objects are available by searching Oracle Fusion Help Application.

Help topics provide information on the import objects, list the reference guides you must consult for information on individual attributes, and provide examples.

You can navigate to help by selecting **Applications Help** from the **Help** menu at the top right of every application page.

There you can view individual help topics or download implementation guides. In the implementation guides, you can find the import topics grouped under the File-Based Data Import heading in the Define Common CRM Configuration task group.

- Oracle Enterprise Repository, which contains reference guides with detailed information on individual attributes.

You can search for the reference guides by name in the repository, which is available at the following URL: <https://fusionappsoer.oracle.com/oer/index.jsp>

To understand what attributes are available for import for each object, follow these steps:

1. Start by searching help using the key word `importing` and the name of the object you are importing.

For example, to get help on importing customers, enter `importing customers`.

Depending on the complexity of the import, your search will return one or more of the following help topic types:

Help Topic Title	Description
Importing <import object name>: Explained	High-level topic that inform the user how each Oracle Fusion business object is structured and explains its relationship to the import object.
<import object name> Import Objects: How They Work Together	This type of topic lists the target objects associated with each import object and provides the names of the Oracle Enterprise Repository reference guides that describe the attributes for those target objects. Target objects are the objects you select when you create the import activity mapping.
Importing <import object name>: Conceptual Example	This type of topic provides an example of how to import the object without detailed steps.
Importing <import object name>: Worked Example	This type of topic walks you through an example step by step.

2. Using the names of the reference guides listed in the <import object name> Import Objects: How They Work Together topic, search for and download the reference guides from the Oracle Enterprise Repository.

These guides include detailed descriptions of attributes and permitted values.

Importing Consumers

Importing Consumers Using File-Based Import: Explained

This topic explains how to prepare and import consumer data from an external data source into Oracle Fusion Applications, using the File-Based Data Import feature. A consumer is a person who intends to purchase goods and services. Within Oracle Fusion Applications, a person with a selling relationship is referred to as a consumer. A consumer import object allows you to import all information about a consumer, such as the consumer profile, consumer address information, consumer contact point information, and consumer contact preference information.

You must consider the following questions before importing consumer information:

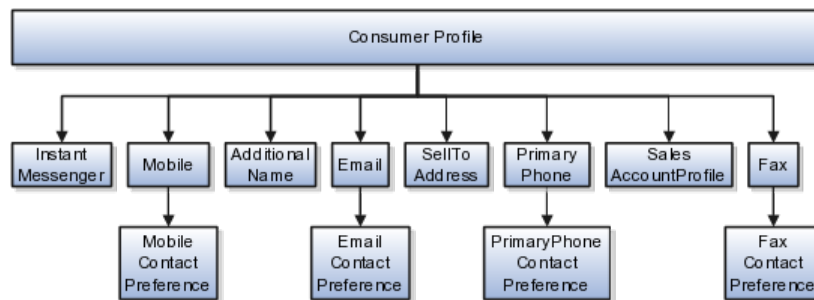
- How does your legacy or source system represent the consumer information compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map them to your existing data values?
- Do you have to customize Oracle Fusion Applications to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

You must understand how your consumer data corresponds with the data in Oracle Fusion Applications to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a consumer.

The Oracle Fusion Applications consumer structure is hierarchical. The root of the hierarchy is the consumer profile, which must exist before you can import lower-level entities, such as e-mail, sell-to address, sales account profile, and fax. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the consumer.

The following figure shows the consumer and its child entities.



The person profile contains basic information about the consumer, such as the first name, last name, and party usage. For each consumer, you can have other information, such as sell-to addresses, phone details, and sales account profile.

Note

All entities referring to contact information, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the consumer for the contact method primary phone.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the consumer.

A good understanding of the import objects and attribute details is critical to preparing your import data. Oracle Fusion Applications attribute details are available for your reference in the Oracle Enterprise Repository. The reference guide files contain descriptions, default values, and validations for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correspond to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

If you have to extend the Oracle Fusion Applications import object to import your legacy or source data, then you must use Oracle Fusion Applications CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Consumers Using File-Based Data Import

For the consumers business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a consumer, you use the Consumer import object. An import activity defines the instructions for processing the import data, including the source file, import mapping from the source file to the Oracle Fusion Applications object and attribute, and the import schedule.

When importing consumer information, you first import the basic person profile information and then the child entities for the consumer. When importing child entities, you must provide the parent reference information for all parent levels for the entity. You must provide the PartyOrigSystem and PartyOrigSystemReference of the consumer when importing contacts for the consumer. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. For example, you first import basic profile details, such as first name, last name, party type, and party usage. You then import other information, such as additional names, e-mail, sell-to address, and primary phone for the consumer.

Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the **Status** column for your

import activity. Alternatively, you can also navigate to the Customer Center work area to view the consumer information that you have imported.

Consumer Import Objects: How They Work Together

The Consumer import object allows you to import consumers, their detailed information, and contacts related to the consumer. This topic describes the Consumer import object and introduces the following:

- Target import object concepts
- Target objects for the Consumer import object
- Target import object attribute
- Target object attribute reference guide files

Consumer Target Import Object Concepts

The Consumer import object imports consumer information, and is split into separate target import objects for organizing the individual attributes of the different aspects of the consumer. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

The target import objects included in the Consumer import object are grouped into information about the consumer. The person profile is the target import object containing attributes to import information about the consumer. When updating an existing consumer with additional information, you must provide the parent reference information for the existing consumer.

To update the information for an existing consumer or to create a consumer record, you can import consumer profile information, addresses, and contact points, such as phone and fax number. The following target import objects are for creating and updating the corresponding consumer information: PersonProfile, PersonAddress, PrimaryPhone, Fax, Mobile, AdditionalName, e-mail, SellToAddress, and SalesAccountProfile.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target import object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes that you want to import do not have an equivalent target object attribute, then review the Oracle Fusion Applications CRM Application Composer extensibility features for the consumer.

Target Import Objects Attributes Resources

To access the reference guide files for the consumer target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Consumer File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
PersonProfile	<p>Includes detailed person information such as person name, relationship type, gender, and marital status.</p> <p>Sample attributes: PersonFirstName, PersonLastName, DateOfBirth, PersonAcademicTitle, and PlaceOfBirth.</p> <p>Reference attributes: PartyId, PartyOrigSystem, PartyOrigSystemReference, RelOrigSystem, RelOrigSystemReference, and RelationshipId.</p>	HZ_IMP_PARTIES_T_Reference
SellToAddress	<p>Includes consumer addresses and party sites information. If party site usage of an address is not defined, then the import process sets it to Sell-To. In case there are multiple addresses without party site usage information, then the import process designates one of the addresses as the Sell-To address.</p> <p>Sample attributes include Address1, Address2, City, Country, County, State, and PostalCode.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes for party sites: SiteOrigSystem and SiteOrigSystemReference, and PartysiteId1.</p> <p>Reference attributes for party site use: SiteuseOrigSystem, SiteuseOrigSystemRef, and PartySiteUseId.</p>	HZ_IMP_PARTYSITES_T_Reference

PrimaryPhone	<p>Indicates the primary phone number of the consumer. If the consumer has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
Mobile	<p>Indicates the mobile phone number of the consumer.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
Fax	<p>Indicates the fax number of the consumer.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
Email	<p>Indicates the e-mail of the consumer.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and StartDate.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference</p>	HZ_IMP_CONTACTPTS_T_Reference

InstantMessenger	<p>Indicates the instant messenger or social networking information of a consumer.</p> <p>Sample attributes: ContactPointPurpose, EndDate, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
SalesAccountProfile	<p>Includes detailed information about a consumer sales account. A sales account is a specific sell-to entity within a given consumer. A consumer can have multiple sales accounts and sales account profiles.</p> <p>Sample attributes: AccountDirectorId, CreationDate, NamedAccountFlag, and AssgnExceptionFlag.</p> <p>Reference attributes: OwnerOrigSystem and OwnerOrigSystemReference.</p>	ZCA_IMP_SALES_ACCOUNTS_Reference
AdditionalName	<p>Indicates alternative name of a consumer.</p> <p>Sample attributes: NameId, PersonFirstName, PersonLastName, and PersonTitle.</p> <p>Reference attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemReference.</p>	HZ_IMP_ADDTNLPARTYNAMES_T_Reference
Classification	<p>Includes classification information for a consumer. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</p> <p>Sample attributes: ClassificationCode, ClassificationCategory, PrimaryFlag, and Rank.</p> <p>Reference Attributes: ClassificationOrigSystem and ClassificationOrigSystemReference.</p>	HZ_IMP_CLASSIFICS_T_Reference

AdditionalIdentifier	<p>Includes detailed information about an additional identifier for a consumer.</p> <p>Sample attributes: IdentifierValue, IdentifierType, Country, and IssuingAuthorityName.</p> <p>Reference attributes: IdentifierValue, AdditionalIdentifierOrigSystem and AdditionalIdentifierOrigSystemRef</p>	HZ_IMP_ADDTNLPARTYIDS_T_Reference
Relationship	<p>Includes information about a relationship between the consumer and another entity, such as an organization or a person.</p> <p>Sample attributes: RelationshipCode, RelationshipType, and DependentFlag.</p> <p>Reference attributes: RelationshipID, RelOrigSystem and RelOrigSystemReference.</p>	HZ_IMP_RELSHIPS_T_Reference
PrimaryPhoneContactPreference	<p>Indicates the consumer's preference about being contacted through phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference Attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
MobileContactPreference	<p>Indicates the consumer's preference about being contacted through mobile phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference Attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

FaxContactPreference	<p>Indicates the consumer's preference about being contacted through fax.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode,</p> <p>Reference Attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
EmailContactPreference	<p>Indicates the consumer's preference about being contacted through mobile phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference Attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

Importing Contacts

Importing Contacts Using File-Based Import: Explained

This topic explains how to prepare and import contact data from an external data source into Oracle Fusion Applications using the File-Based Data Import feature. A contact is a person party related to a customer, and the contact object contains information that identifies the contact and offers the contact points of the contact. Contact points can be geographical addresses, phone numbers, e-mail IDs, URLs, messenger IDs, and so on. The contact object also contains contact preference information for the contact.

You must create or import contacts before you can associate them with customer objects. You must consider the following questions before importing contact information:

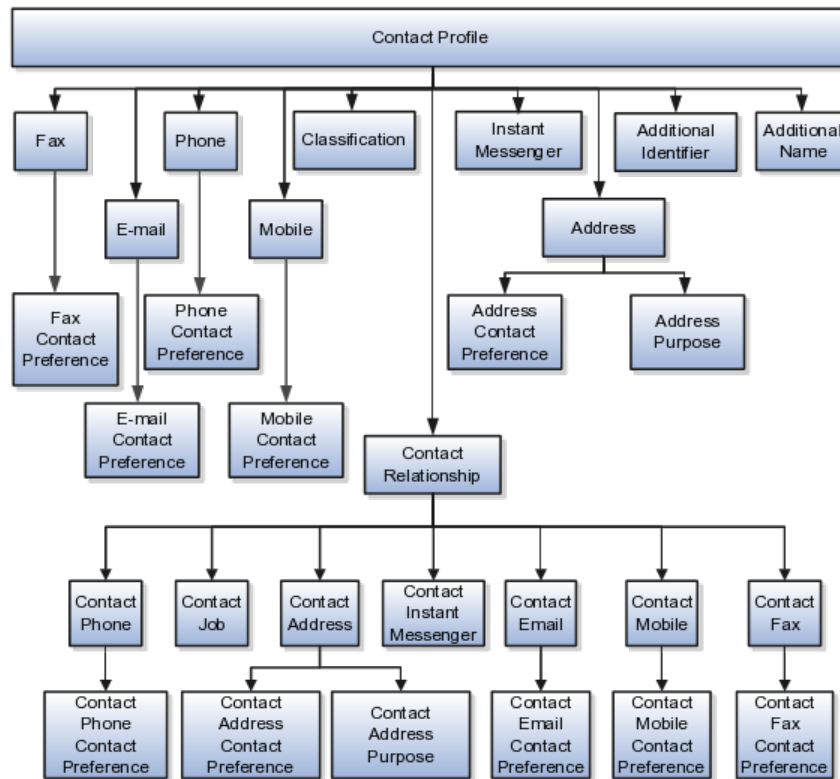
- How does your legacy system or source system represent the contact compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map to your data values?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your contact data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a contact.

In Oracle Fusion Applications, one table stores the contact information and another table optionally stores contact point details for that contact. A contact

point is an identified means to contact a party. Contact points can be phone numbers, e-mail IDs, Web site addresses, or instant messenger IDs. The following figure shows the contact object and its child entities.



The contact profile contains basic information about the contact, such as the contact name, party type, and party usage. For each contact, you can assign classifications, phone details, and additional names. If the contact is a consumer, then the contact relationship of the consumer includes other child entities that capture information about the consumer contact, such as contact job, contact phone, contact address, and contact e-mail.

Note

All contact-related entities, such phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the consumer for the contact method phone.

Import Objects for the Contact

To facilitate the import of contacts, Oracle Fusion Applications incorporate the structure of the contact business object into import objects. The import object for contacts is Contact.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the contact.

A good understanding of the attribute details of the import objects is critical to preparing your import data. The reference guide files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correlate to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

Import Object	Related Import Object Topic
Contact	Contact Import Objects: How They Work Together

Hint: You can use the keyword importing contacts to search for related topics in Oracle Fusion Applications Help.

Extensible Attributes

If you need to extend the Oracle Fusion Applications object to import your legacy or source data, you must use Oracle Fusion CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Contacts Using File-Based Data Import

For the Contact business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new contact, you import the Contact object. You must be assigned the master Data Management Administrator job role to access and submit the import activities for contacts.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. You can also navigate to the Customer Center work area to view the contact information that you have imported.

Contact Import Objects: How They Work Together

You can import all your contact information using the Contact import object. This topic describes this import object and introduces the following

- Target import object concepts
- Target objects for the Contact import object
- Target import object attributes

- Target import object attribute reference guide files

Target Import Object Concepts

The Contact import object is used to import a contact's profile information, such as first name, last name, address, and so on, and contact point information, such as phone numbers, fax, e-mail IDs, and so on. The Contact import object is split into separate target import objects for organizing the individual attributes of the different aspects of the contact's profile and contact points. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Contact Target Import Objects

The target import objects included in the Contact import are grouped into information about the contact and the contact's relationship with other parties. The contact profile is the target import object containing attributes to import information about the contact. You can have multiple contact relationships associated with a contact. There are multiple target import objects that include attributes to import contacts and their related information.

When updating an existing contact, you must provide the parent reference information of the existing contact. When importing contact profile or contact point information for a contact, you must provide relationship reference information in addition to the parent reference. You must provide this information because a contact can have multiple relationships with an organization, such as employee or board member. When importing information about a contact, you must refer to the specific relationship that you want to import information for. For example, you must specify whether you want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for a contact relationship, then the import process creates a new relationship.

To update the information for an existing contact or to create a contact record, you can import contact profile information, addresses, and contact points. The following target import objects are for creating and updating contact information: PersonProfile, Fax, PrimaryPhone, Url, SalesAccountProfile, Classification, AdditionalName, Relationship, and AdditionalIdentifier.

To update or create a contact point, use the following target import objects: ContactRelationship, ContactJob, ContactAddress, ContactEmail, ContactPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-point-related entities, such as phone or e-mail, include a child entity that captures the contact preference. For example, the ContactPhoneContactPreference entity captures the contact preference of the primary phone contact method.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping

task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes that you want to import do not have an equivalent for the target object attribute, then review the Oracle Fusion CRM Application Composer Extensibility features for the marketing response.

Target Import Objects Attributes Resources

To access the reference guide files for the response's target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Response File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
PersonProfile	Includes information that identifies a contact. Sample attributes: PersonFirstName, PersonLastName, DateOfBirth, gender, HouseholdIncome, and PlaceOfBirth Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.	HZ_IMP_PARTIES_T_Reference
Fax	Indicates the fax number of the contact. Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber. Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.	HZ_IMP_CONTACTPTS_T_Reference
FaxContactPreference	Indicates the contact's preferences about being contacted through Fax. Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode. Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.	HZ_IMP_CONTACTPREFS_Reference

Phone	<p>Indicates the phone number of the contact. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number. The attributes of PrimaryPhone, Fax, and URL are the same. However, the value of the attribute ContactPointType is different for each of these contact point related entities.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
PhoneContactPreference	<p>Indicates the contact's preferences about being contacted by phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
Email	<p>Indicates the e-mail of the contact.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and StartDate.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
EmailContactPreference	<p>Indicates the contact's contact preferences about being contacted through e-mail.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference

InstantMessenger	<p>Indicates the instant messenger information of a contact.</p> <p>Sample attributes: ContactPointPurpose, EndDate, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
Mobile	<p>Indicates the mobile number of a contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
MobileContactPreference	<p>Indicates the contact's preferences about being contacted through mobile phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
Classification	<p>Includes classification information for a contact. Classification allows you to categorize entities, such as parties, projects, tasks, and orders as hierarchies.</p> <p>Sample attributes: ClassificationCode, ClassificationCategory, PrimaryFlag, and Rank.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CLASSIFICS_T_Reference

Address	<p>Indicates the address of a contact. If the contact has multiple addresses, then the first address is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, State, and PostalCode.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_PARTYSITES_T_Reference
AddressContactPreference	<p>Indicates the contact's preferences about being contacted at the primary address.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
AdditionalName	<p>Indicates the alternative name of a contact.</p> <p>Sample attributes: Additional Name, Additional Name Type, and GlobalPreferredNameFlag.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_ADDTNLPARTYNAMES_T_Reference
AdditionalIdentifier	<p>Includes the basic information about an additional identifier for the contact.</p> <p>Sample attributes: IdentifierValue, IdentifierType, Country, and IssuingAuthorityName.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_ADDTNLPARTYIDS_T_Reference

ContactRelationship	<p>Includes information about a relationship between the contact and other parties.</p> <p>Sample attributes: RelationshipCode, RelationshipType, and DependentFlag.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_RELSHIPS_T_Reference
ContactEmail	<p>Indicates the e-mail contact point of the contact.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and StartDate.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactEmailContactPreference	<p>Indicates the contact's preferences about being contacted through the e-mail contact point.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactFax	<p>Indicates the number of the fax contact point of the contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, and PhoneExtension.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference

ContactFaxContactPreference	<p>Indicates the contact's preference about being contacted through the fax contact point.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactInstantMessenger	<p>Captures information about a contact's instant messenger contact point.</p> <p>Sample attributes: ContactPointPurpose, EndDate, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactMobile	<p>Indicates the number of a contact's mobile contact point.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactMobileContactPreference	<p>Indicates the contact's preferences about being contacted through the mobile phone contact point.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference

ContactPhone	<p>Indicates the number of the contact's phone contact point. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactPhoneContactPreference	<p>Indicates the contact's preferences about being contacted through the phone contact point.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactAddress	<p>Indicates the address of a contact's address contact point. If the contact has multiple addresses, then the first address is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, State, and PostalCode.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_PARTYSITES_T_Reference

ContactAddressContactPreference	<p>Indicates the contact's preferences about being contacted at the address contact point.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactJob	<p>Includes the job information of a contact.</p> <p>Sample attributes: Comments, ContactNumber, Department, DepartmentCode, JobTitle, and JobTitleCode.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_CONTACTS_T_Reference

Importing Country Structures

Importing Country Structures Using File-Based Import: Explained

This topic explains how to prepare and import country structure data from an external data source into Oracle Fusion Applications using the File-Based Data Import feature. A country structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States has the geography type of State at the top, followed by the County, then the City, and finally the Postal Code.

You can use the country structure to set up the following:

- The relationships between geographies within a country
- The types of geographies that you can define for a country

Consider the following questions when importing your data:

- How does your legacy system or source system represent the country structure compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map to your data values?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?

- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your country structure data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a country structure.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country's structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Country Structure

To facilitate the import of country structures, Oracle Fusion Applications incorporate the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the country structure.

A good understanding of the attribute details of the import objects is critical to preparing your import data. For information about the Oracle Fusion Applications attributes, see the Oracle Enterprise Repository. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correlate to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference file names and locations that you need to complete this task, see the following table.

Import Object	Related Import Object Topic
Country Structure	Country Structure Import Objects: How They Work Together

Extensible Attributes

If you need to extend the Oracle Fusion Applications object to import your legacy or source data, you must use Oracle Fusion CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The

corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Country Structures Using File-Based Data Import

For the country structure business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new country structure, you import the Country Structure object.

You must be assigned the Master Data Management Administrator job role to access and submit the import activities for country structures.

Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity.

Country Structure Import Objects: How They Work Together

This topic describes the Country Structure import object. You use the Country Structure import object when you submit a file-based import activity to import your country structure information. This topic introduces the following:

- Target import object concepts
- Target objects for the Country Structure import object
- Target import object attributes
- Target object attribute reference guide files

Target Import Object Concepts

The Country Structure import object is used to import a country structure hierarchy, including details, such as geography type, geography type name, parent geography type, geography level numbers, and so on. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Country Structure Target Import Objects

The Country Structure import object contains one target import object that organizes the individual attributes of the different aspects of the geography

structure. When updating an existing country structure, you must provide the parent reference information of the existing country structure. This reference information connects the imported geography structure to the existing one. Use the ImpGeoStructureLevel target import object to create and update country structure information.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import does not have an equivalent target object attribute, then review the Oracle Fusion CRM Application Composer extensibility features for country structures.

Target Import Objects Attributes Resources

To access the reference guide files for the country code's target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference guide files that are available from the Documentation tab for the Country Code File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
ImpGeoStructureLevel	<p>Contains information that specifies a country's geography structure.</p> <p>Sample attributes: GeographyType, GeographyTypeName, LevelNumber, and ParentGeographyType.</p> <p>Reference attribute: CountryCode</p>	HZ_IMP_GEO_STRUCTURE_LEVELS_Reference

Importing Customer Hierarchies

Importing Customer Hierarchy Using File-Based Import: Explained

This topic describes how to import a customer hierarchy and customer hierarchy members. A customer hierarchy captures the hierarchical relationships that a customer has with other customers. A customer hierarchy consists of hierarchy members. A customer hierarchy can be used to process payments from one customer and apply them to another customer in the same hierarchy. It can also be used to create the revenue roll-up report that rolls up revenue numbers from opportunities for all customers in a hierarchy.

You must consider the following questions before importing customer hierarchy information:

- Are all members of the customer hierarchy in Oracle Fusion Applications?
- Do I have to import additional hierarchy members to complete the hierarchy?
- How does your legacy system or source system represent the promotion compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map your existing data values to the Customer import object?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

To import a customer hierarchy, you must:

1. Import customers who must be a part of the hierarchy.
2. Import a hierarchy definition that includes the hierarchy structure and the root node.
3. Import members from the existing customers into the hierarchy.

Comparing Business Object Structures

Before you can import your legacy or source system customer hierarchy data, you must first analyze the data and see how it corresponds to the customer object structure of Oracle Fusion Applications. You must understand how Oracle Fusion Applications represent the structure of the data for a customer hierarchy.

The Oracle Fusion Applications customer hierarchy object structure is hierarchical. At the top level is the customer hierarchy, which must exist before you can import customer hierarchy members. The customer hierarchy and customer hierarchy member contain information about the customer hierarchy

and customer hierarchy nodes, such as the hierarchy type, hierarchy code, hierarchy name, and hierarchy version.

Comparing Business Object Data

After you understand the structure of the data, then compare the detailed attribute values of your data with the Oracle Fusion Applications data. Each import object is a collection of attributes organized to assist you when mapping your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the customer hierarchy.

A good understanding of the import objects and attribute details is critical to preparing your import data. Oracle Fusion Applications attribute details are available for your reference in the Oracle Enterprise Repository. The reference guide files contain descriptions, default values, and validations for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correspond to a choice list in Oracle Fusion Applications, then the validation information for that attribute will provide the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

If you want to extend the Oracle Fusion Applications import object to import your legacy or source data, then you must use Oracle Fusion Applications CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing.

The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Customer Hierarchy Using File-Based Data Import

For the customer hierarchy business object, you must use the File-Based Data Import feature. You prepare XML or text source data files, such as CSV, in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. The Define File-Based Data Import Setup and Maintenance task list includes the tasks required to configure the import objects, to create source file mappings, and to schedule the import activities. You submit file-based import activities for each import object.

An import activity defines the instructions for processing import data, including the source file, import mapping from the source file to the Oracle Fusion Applications object and attribute, and the import schedule.

When importing customer hierarchy information, you first import the customer hierarchy information and then the customer hierarchy members for the

customer hierarchy. When importing customer hierarchy members, you must provide the parent reference information that refers to the customer hierarchy of the member.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. You can view the list of import activities from the Manage Import Activities page. You can verify the status of the import activity by clicking the **Status** column for your import activity. Alternatively, you can also navigate to the Manage Hierarchies task from Setup and Maintenance to view the customer hierarchy information that you have imported.

Customer Hierarchy Import Objects: How They Work Together

The Customer Hierarchy import object imports customer hierarchy and customer hierarchy members. This topic describes the Customer Hierarchy object and introduces the following:

- Target import object concepts
- Target objects for the Customer Hierarchy import object
- Target import object attributes
- Target import object attribute reference guide files for evaluating and mapping source file data

Customer Hierarchy Target Import Objects Concepts

The Customer Hierarchy import object imports customer hierarchy and customer hierarchy members. The Customer Hierarchy and Customer Hierarchy Members import objects are split into separate target import objects for organizing the individual attributes for the different aspects of the customer hierarchy. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Target Import Objects

The target import objects in the customer hierarchy import object are grouped into information about the customer hierarchy and information about the customer hierarchy members. The CustomerHierarchy is the target import object containing attributes to import information about the customer hierarchy, and the CustomerHierarchyMember is the target import object containing attributes to import information about the customer hierarchy members. You can have multiple hierarchy members associated with a customer hierarchy.

To import a customer hierarchy, you must do the following:

1. Import customers who must be part of the hierarchy.
2. Import the customer hierarchy definition that includes the hierarchy structure and the root node.

3. Import existing customers as customer hierarchy members into the hierarchy.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes that you want to import do not have an equivalent target object attribute, then review the Oracle Fusion Applications CRM Application Composer extensibility features for the customer hierarchy.

Target Import Objects Attributes Resources

To access the reference guide files for the customer hierarchy target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Customer Hierarchy File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
CustomerHierarchy	Customer hierarchy definition and root node of the customer hierarchy	HZ_IMP_HIERARCHIES_T_Reference
CustomerHierarchyMember	Customer hierarchy members	HZ_IMP_HIERARCHY_NODES_Reference

Importing Customers

Importing Customers Using File-Based Import: Explained

This topic explains how to prepare and import customer data from an external data source into Oracle Fusion Applications, using the File-Based Data Import feature.

A customer is an entity with whom you have a selling relationship. Oracle Fusion Applications refer to sales account and sales prospects collectively as a Customer. A customer business object allows you to capture all information about a customer, such as the customer profile, customer contact relationships information, and customer contact points.

A sales account is a specific sell-to entity within a given customer. You can create leads and opportunities for sales accounts. An entity with a sales account in Oracle Fusion Applications can be one of the following:

- Sales Prospect: A prospective entity or person who does not have a sell-to address. This entity is used to define leads.
- Customer: A sales account or sales prospects.
- Legal Entity: A legal entity is a party that can enter into legal contracts or a business relationship. It can be sued if it fails to meet contractual obligations.

You can use the Customer import object to import sales accounts or sales prospects of the type organization. You can use the Group Customer import object to import customers of the type group, and the Consumer import object to import customers of the type person.

You must consider the following questions when importing your data:

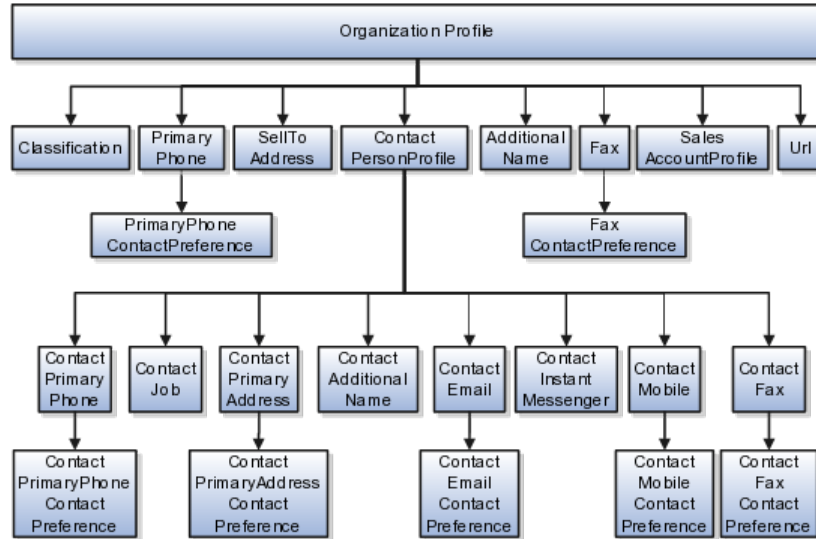
- Did you identify the records that should be imported as sales prospects, and the records that should be imported as sales accounts?
- How does your legacy or source system represent the customer information compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map your existing data values to the Customer import object?
- Do you have to customize Oracle Fusion Applications to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

Before you can import your customer data from the legacy or source system, you must first analyze the data and see how it corresponds to the customer object structure of Oracle Fusion Applications. You must understand how Oracle Fusion Applications represent the structure of the data for a customer.

The Oracle Fusion Applications Customer import object structure is hierarchical. The root of the hierarchy is the organization profile, which must exist before you can import lower-level components, such as classifications, contacts, and sales account profiles. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the customer.

The following figure shows the customer object and its child entities.



The organization profile contains basic information about the customer, such as the customer name and party usage. For each customer, you can assign classifications, contacts, sell to addresses, phone details, and additional names. The contact of the customer, in turn, includes other child entities that capture information about the contact, such as contact job, contact primary phone, contact primary address, and contact e-mail.

Note

All contact entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the customer for the contact method primary phone.

Comparing Business Object Data

After you understand the structure of the data, the next step is to compare the detailed attribute values of your data with the Oracle Fusion Applications data. Each import object is a collection of attributes organized to assist you when mapping your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the customer.

A good understanding of the import objects and attribute details is critical to preparing your import data. The reference guide files contain descriptions, default values, and validations for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correspond to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

To extend the Oracle Fusion Applications object to import your legacy or source data, you must use Oracle Fusion CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import

object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Customers Using File-Based Data Import

For the customer business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Import Setup and Maintenance task list includes the tasks required to configure the import objects, to create source file mappings, and to schedule the import activities. You submit file-based import activities for each import object. An import activity defines the instructions for processing import data including the source file, import data mapping from the source file to the Oracle Fusion Applications object and attribute, and the import schedule.

When importing customer information, you first import the basic customer profile information, followed by the child entities for the customer. When importing child entities, you must provide the parent reference information for all parent levels for the entity. For example, you first import basic profile details, such as customer name, party type, and party usage. You then import contacts and contact information, such as phone, address, contact points, and fax for the customer. You must provide the PartyOrigSystem and PartyOrigSystemReference of the customer when importing contacts for the customer. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. When importing contact information for a customer, you must provide the relationship reference information. This information is required because a contact can have multiple relationships with a customer.

Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the **Status** column for your import activity. Alternatively, you can also navigate to the Customer Center work area to view the customer information that you have imported.

Customer Import Objects: How They Work Together

The Customer import object allows you to import customers, their detailed information, and contacts related to the customer. This topic describes the Customer import object. It introduces the following:

- Target import object concepts
- Target objects for the Customer import object
- Target import object attribute
- Target import object attribute reference guide files for evaluating and mapping source file data

Customer Target Import Object Concepts

The Customer import object imports customer information and contacts related to customer. The Customer import object is split into separate target import

objects for organizing the individual attributes for the different aspects of the customer and customer contact. Before you import customer information, you must understand how the target objects are related and what attributes are included in each object. This understanding helps you create the appropriate mapping between the source data in your import file and the target attributes.

The target import objects included in the Customer import object are grouped into information about the customer and the customer contact. The organization profile is the target import object containing attributes to import information about the customer. You can have multiple contacts associated with a customer. The Customer import object has multiple target import objects that include attributes to import customer contacts and their related information.

When updating an existing customer with additional information, you must provide the parent reference information for the existing customer. When importing customer contacts or contact information for a customer, you must provide relationship reference information in addition to the parent reference. This information is required because a contact can have multiple relationships with an organization, such as an employee or board member. When importing information about a contact you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for a customer-contact relationship, then the import process creates a relationship.

To update the information for an existing customer or to create a customer record, you can import customer profile information, addresses, and contact points, such as a phone and fax. The following target import objects are for creating and updating the corresponding customer information: OrganizationProfile, Fax, PrimaryPhone, Url, SalesAccountProfile, Classification, AdditionalName, Relationship, SellToAddress, and AdditionalIdentifier.

To update or to create a customer contact, use the following target import objects: ContactPersonProfile, ContactJob, ContactPrimaryAddress, ContactEmail, ContactPrimaryPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import do not have an equivalent target object attribute, then review the Oracle Fusion Applications CRM Application Composer extensibility features for the customer.

Target Import Objects Attributes Resources

To access the reference guide files for the customer target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Customer File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
OrganizationProfile	<p>Includes detailed customer information, such as, organization name and organization type.</p> <p>Sample attributes: OrganizationSize, UniqueNameAlias, CeoName, and CertificationLevel.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_PARTIES_T_Reference
PrimaryPhone	<p>Indicates the primary phone number of the customer. If the customer has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
Classification	<p>Includes classification information for a customer. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</p> <p>Sample attributes: ClassificationCode, ClassificationCategory, PrimaryFlag, and Rank.</p> <p>Reference attributes: ClassificationOrigSystem and ClassificationOrigSystemReference.</p>	HZ_IMP_CLASSIFICS_T_Reference

AdditionalName	<p>Indicates alternative name of a customer.</p> <p>Sample attributes: Additional Name, Additional Name Type, and GlobalPreferredNameFlag.</p> <p>Reference attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemRefere</p>	HZ_IMP_ADDTNLPARTYNAMES_T_Reference
Fax	<p>Indicates the fax of the customer.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
FaxContactPreference	<p>Indicates the customer preference about being contacted through Fax.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
Url	<p>Indicates the Url of the customer organization.</p> <p>Sample attributes: ContactPointType, PrimaryFlag, Url, and WebType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference

SalesAccountProfile	<p>Includes detailed information about a customer sales account. A sales account is a specific sell-to entity within a given customer. A customer can have multiple sales accounts and sales account profiles.</p> <p>Sample attributes: AccountDirectorId, NamedAccountFlag, and AssgnExceptionFlag.</p> <p>Reference attributes: OwnerOrigSystem and OwnerOrigSystemReference.</p>	ZCA_IMP_SALES_ACCOUNTS_Reference
SellToAddress	<p>Includes customer addresses and party sites information. If party site usage of an address is not defined, then the import process sets it to Sell-To. If there are multiple addresses without party site usage information, then the import process designates one of the addresses as the Sell-To address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, and State.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes for party sites: SiteOrigSystem, SiteOrigSystemReference, and PartysiteId1.</p> <p>Reference attributes for party site use: SiteuseOrigSystem, SiteuseOrigSystemRef, and PartySiteUseId.</p>	HZ_IMP_PARTYSITES_T_Reference
AdditionalIdentifier	<p>Includes detailed information about an additional identifier for the customer.</p> <p>Sample attributes: IdentifierValue, IdentifierType, Country, and IssuingAuthorityName.</p> <p>Reference attributes: IdentifierValue, AdditionalIdentifierOrigSystem, and AdditionalIdentifierOrigSystemRef</p>	HZ_IMP_ADDTNLPARTYIDS_T_Reference

Relationship	<p>Includes information about a relationship between the customer and another entity, such as an organization or a person. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</p> <p>Sample attributes: RelationshipCode, RelationshipType, and DependentFlag.</p> <p>Reference attributes: RelationshipID, RelOrigSystem and RelOrigSystemReference.</p>	HZ_IMP_RELSHIPS_T_Reference
ContactPersonProfile	<p>Includes detailed information, such as, person name, relationship type, and marital status, of a customer contact.</p> <p>Sample attributes: Gender, MaritalStatus, PartyType, PartyUsageCode, PersonFirstName, and PersonLastName.</p> <p>Reference attributes: PartyID, PartyOrigSystem, PartyOrigSystemReference, RelOrigSystem, RelOrigSystemReference, and RelationshipId.</p>	HZ_IMP_RELSHIPS_T_Reference
ContactAdditionalName	<p>Indicates alternative name of a customer contact.</p> <p>Sample attributes: NameId, PersonFirstName, PersonLastName, and PersonTitle.</p> <p>Reference attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemRefere</p>	HZ_IMP_ADDTNLPARTYNAMES_T_Reference
ContactEmail	<p>Indicates the e-mail of the customer contact.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and StartDate.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference

ContactInstantMessenger	<p>Indicates the instant messenger or social networking information of a customer contact.</p> <p>Sample attributes: ContactPointPurpose, EndDate, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactPrimaryAddress	<p>Indicates the primary address of a customer contact. If the contact has multiple addresses, one of the addresses is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, and State.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes or party sites: SiteOrigSystem, SiteOrigSystemReference, and PartysiteId1.</p>	HZ_IMP_PARTYSITES_T_Reference
ContactFax	<p>Indicates the fax of the customer contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, and PhoneExtension.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactJob	<p>Includes job information of a customer contact.</p> <p>Sample attributes: Comments, ContactNumber, Department, DepartmentCode, and JobTitleCode.</p> <p>Reference attributes: RelationshipID, RelOrigSystem and RelOrigSystemReference</p>	HZ_IMP_CONTACTS_T_Reference

ContactMobile	<p>Indicates the mobile number of a customer contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactPrimaryPhone	<p>Indicates the primary phone number of the customer contact. If the customer has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
PrimaryPhoneContactPreference	<p>Indicates the customer contact preference about being contacted through phone.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactEmailContactPreference	<p>Indicates the customer contact preference about being contacted through e-mail.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

ContactFaxContactPreference	<p>Indicates the customer contact preference about being contacted through Fax.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactMobileContactPreference	<p>Indicates the customer contact preference about being contacted through mobile phone.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactPrimaryAddressContactPre	<p>Indicates the customer contact preference about being contacted at the primary address.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactPrimaryPhoneContactPrefe	<p>Indicates the customer contact preference about being contacted through the primary phone number.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

Importing Employee Resources

Importing Employee Resources Using File-Based Import: Explained

This topic explains how to prepare and import employee resource data from an external data source into Oracle Fusion Applications using the File-Based Data Import feature.

Employee resources are employees within the deploying company who can be assigned work objects. You must create or import employee resources before you can associate them with resource organizations, resource teams, or work objects. You enter your employee resource information using the Setup and Maintenance work area, Manage Resources task, or you can import data to create new or update existing employee resources.

Consider the following questions when importing your data:

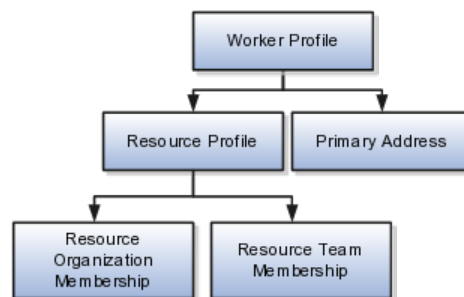
- How does your legacy system or source system represent the employee resource compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map to your data values?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your employee resource data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for an employee resource.

In Oracle Fusion Applications, one table stores the employee resource definition and other tables optionally store profile details for that employee resource. Profile details for a resource include information about an employee resource's organization and team memberships.

The following figure illustrates the structure of the Employee Resource object.



The worker profile contains basic information about the employee resource, such as the employee resource's primary address and resource profile details.

Import Objects for the Employee Resource

To facilitate the import of employee resources, Oracle Fusion Applications incorporate the structure of the employee resource into import objects. The import object for the employee resource is Employee Resource.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the employee resource.

A good understanding of the attribute details of the import objects is critical to preparing your import data. The reference guide files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correlate to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

Import Object	Related Import Object Topic
Employee Resource	Employee Resource Import Objects: How They Work Together

Hint: You can use the keyword importing employee resources to search for related topics in Oracle Fusion Applications Help.

Extensible Attributes

If you need to extend the Oracle Fusion Applications object to import your legacy or source data, you must use Oracle Fusion CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Employee Resources Using File-Based Data Import

For the employee resource business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new employee resource, you import the Employee Resource import object.

You must be assigned the Master Data Management Administrator job role to access and submit the import activities for employee resources.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. Users with either the Master Data

Management Administrator job role can also navigate to the Manage Resources work area to view the imported employee resources.

Employee Resource Import Objects: How They Work Together

Employee resources are employees to whom you can assign work objects, for example, service agents, sales managers, and so on. You can import all your employee resource information using the Employee Resource import object. This topic describes this import object.

This topic introduces the following:

- Target import object concepts
- Target objects for the Employee Resource import object
- Target import object attributes
- Target import object attribute reference guide files

Target Import Object Concepts

The Employee Resource import object is used to import an employee resource's worker profile information, such as first name, last name, address, and so on, and the employee resource's resource profile information, such as resource organization and resource team memberships. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Employee Resource Target Import Objects

The target import objects included in the Employee Resource import object are grouped into information about the employee resource's profile and the employee resource's address information. The Worker profile is the target import object containing attributes to import information about the employee resource.

When updating an existing employee resource, you must provide the parent reference information of the existing employee resource. When importing an employee resource's resource team information, you must provide relationship reference information in addition to the parent reference, because an employee resource can belong to more than one team simultaneously.

To update the information for an existing employee resource or to create an employee resource record, you can import employee resource profile information and addresses. The following target import objects are for creating and updating employee resource information: WorkerProfile, ResourceProfile, and PrimaryAddress.

To update or create an employee resource's resource profile, use the following target import objects: ResourceOrganizationMembership and ResourceTeamMembership.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import do not have an equivalent target object attribute, then review the Oracle Fusion CRM Application Composer extensibility features for the marketing response.

Target Import Objects Attributes Resources

To access the reference guide files for the employee resource's target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Employee Resource File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
WorkerProfile	<p>Contains information that identifies an employee.</p> <p>Sample attributes: ResourceProfile and PrimaryAddress.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HRC_LOADER_BATCH_LINES_Reference
ResourceProfile	<p>Contains information about a resource.</p> <p>Sample attributes: PartyId, FirstName, LastName, Department, and JobCode.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_RESOURCES_Reference
PrimaryAddress	<p>Indicates the address of a resource. If the resource has multiple addresses, then the first address is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, State, and PostalCode.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_PARTYSITES_T_Reference

ResourceOrganizationMembership	<p>Contains the employee resource's resource organization membership information.</p> <p>Sample attributes: RequestId, PartyId, ParentOrganizationName, OrganizationId, OrganizationName, and OrganizationUsage.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_GROUP_MEMBERS_Reference
ResourceTeamMembership	<p>Contains the employee resource's resource team membership information.</p> <p>Sample attributes: RequestId, TeamMemberId, TeamId, and TeamNumber.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_TEAM_MEMBERS_Reference

Importing Geographies

Importing Geographies Using File-Based Import: Explained

This topic describes the tasks you must perform to import geography information. A geography is any region with a boundary around it, regardless of its size. It might be a state, a country, a city, a county, or a ward. You must create or import geographies before you can associate them with custom zones and addresses.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map to your data values?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your geography data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a geography.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in

the country's structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Geography

To facilitate the import of geographies, Oracle Fusion Applications incorporate the structure of the geography into import objects. The import object for the geography is ImpGeography.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the geography.

A good understanding of the attribute details of the import objects is critical to preparing your import data. For information about the Oracle Fusion Applications attributes, see the Oracle Enterprise Repository. The reference files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correlate to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference file names and locations that you need to complete this task, see the following table.

Import Object	Related Import Object Topic
ImpGeography	Geography Import Objects: How They Work Together

Hint: You can use the keyword importing geographies to search for related topics in Oracle Fusion Applications Help.

Extensible Attributes

Oracle Fusion Applications do not support extensible attributes for geographies. You can only import data for attributes provided by Oracle Fusion Applications.

Importing Geographies Using File-Based Data Import

For the geography business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new geography, you import the Geography object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for geographies.

When importing geography information, you must provide the parent reference information for all parent levels for the entity.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Geographies work area to view the imported geographies.

Geography Import Objects: How They Work Together

This topic describes the Geography import object. You use the Geography import object to import geography information.

This topic introduces the following:

- Target import object concepts
- Target objects for the Geography import object
- Target import object attributes
- Target import object attribute reference guide files

Target Import Object Concepts

The Geography import object is used to import geography hierarchy information to create or update the geography data of a country. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Geography Target Import Objects

The target import objects in the Geography import object contain information about the geography hierarchy. When updating an existing geography, you must provide the parent reference information of the existing geography, which connects the geography to the country of which it is a part.

Use the ImpGeography target import object to create and update geography information.

Note

Before you import geography data for a country, you must define the country's geography structure.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Target Import Objects Attributes Resources

To access the reference guide files for the geography's target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, *Importing Master Reference Geography Data*, on the Oracle Support site.

The following table lists the reference guide files that are available from the Documentation tab for the Geography File-Based Data Import asset.

Target Import Object	Description	Attribute Reference Guide File Names
ImpGeography	<p>Contains information that captures a country's geography hierarchy details.</p> <p>Sample attributes: CountryCode, GeoDataProvider, GeographyType, PrimaryGeographyCode, PrimaryGeographyCodeType, and PrimaryGeographyName.</p> <p>Reference attribute: CountryCode</p>	HZ_IMP_GEOGRAPHIES_T_Reference

Importing Group Customers

Importing Group Customers Using File-Based Import: Explained

This topic explains how to prepare and import group customer data from an external data source into Oracle Fusion Applications, using the File-Based Data Import feature.

A group customer is a party consisting of a collection of persons and organizations. A group, similar to a person or organization, can enter into a business relationship with other parties and can be assigned to opportunities and leads. A group can have members of party type person or organization. You can maintain group customers information in Oracle Fusion Applications CRM to create leads and opportunities for a group, or capture information about a group to know them better. However, maintaining this information may not let you assign all sales and marketing activities to a group.

You must consider the following questions before importing group customer information:

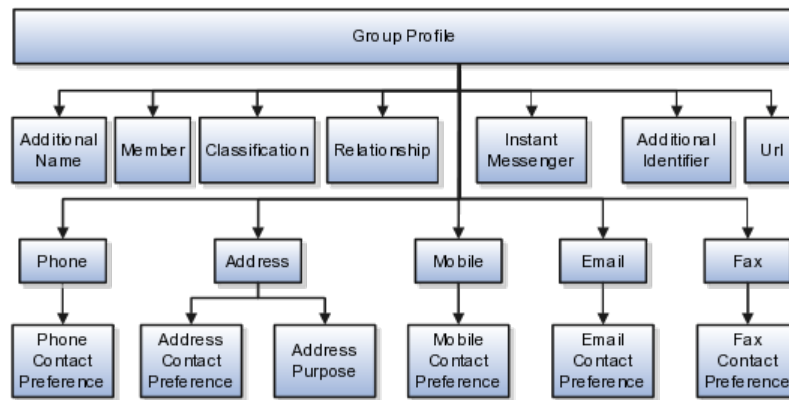
- How does your legacy or source system represent the group customer information compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map your existing data values to the Group Customer import object?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

You must understand how your group customer data corresponds with the data in Oracle Fusion Applications to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a group customer.

The Oracle Fusion Applications Group Customer structure is hierarchical. At the top level is the group profile, which must exist before you can import lower-level components, such as classifications, additional names, additional identifier, and members. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the group customer.

The following figure shows the Group Customer import object and its child entities.



The group customer profile contains basic information about the group customer, such as the group name, group type, and the group-party usage. For each group customer, you can assign classifications, members, relationships, additional identifier, and additional names.

Note

All entities referring to contact information, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the group for the phone contact method.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the group customer.

A good understanding of the import objects and attribute details is critical to preparing your import data. Oracle Fusion Applications attribute details are available for your reference in the Oracle Enterprise Repository. The reference guide files contain descriptions, default values, and validations for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correspond to a choice list in Oracle Fusion Applications, then the validation information for that attribute will provide the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

If you want to extend the Oracle Fusion Applications object data to import your legacy or source data, then you must use Oracle Fusion Applications CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Group Customers Using File-Based Data Import

For the group customer business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. An import activity defines the instructions for processing import data, including the source file, import mapping from the source file to the Oracle Fusion Applications object and attribute, and the import schedule.

When importing group customer information, you must first import the basic group customer profile information, followed by the child entities for the group customer. When importing child entities, you must provide the parent reference information for all parent levels for the entity. You must provide the PartyOrigSystem and PartyOrigSystemReference of the group customer when importing contacts for the group customer. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. For example, you first import basic profile details, such as first name, last name, party type, and party usage. You then import contact information, such as phone, address, contact points, and fax for the group customer.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which you can use to verify imported data. You can view the list of import activities from the Manage Import Activities page. You can verify the status of the import activity by clicking the **Status** column for your import activity. Alternatively, you can also navigate to the Party Center work area to view the customer information that you have imported.

Group Customer Import Objects: How They Work Together

The Group Customer import object allows you to import group customers, their basic information, and members of the group. This topic describes the Group Customer import object and introduces the following:

- Target import object concepts
- Target objects for the Group Customer import object

- Target import object attributes
- Target import object attribute reference guide files for evaluating and mapping source file data

Group Customer Target Import Objects Concepts

The Group Customer import object imports group customers and its members. The Group Customer import object is split into separate target import objects for organizing the individual attributes for the different aspects of the group customer. To map the source data in your import file to the target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Group Customer Target Import Objects

The target import objects in the Group Customer import object are grouped into information about the group customer and information about the member. The group profile is the target import object containing attributes to import information about the group customer. You can have multiple members associated with a group customer. You can assign only an organization or a person as a member of a group. You cannot assign a group as a member of another group.

When updating an existing group customer with additional information, you must provide the parent reference information for the existing group customer. When importing contacts or contact information for a group customer, you must provide relationship reference information in addition to the parent reference. When importing information about a member, you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for a relationship, then the import process will create a relationship.

To update the information for an existing group customer or to create a group customer record, you can import group customer profile information, addresses, and contact points, such as phone and fax. The following target import objects are for creating and updating the group customer information: GroupProfile, Fax, Mobile, Phone, e-mail, InstantMessenger, Relationship, Member, Address, Classification, AdditionalName, and AdditionalIdentifier.

All contact-related entities, such as phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the group customer for the contact method primary phone. Additionally, the Address import object for a group customer includes another child entity, AddressPurpose, that captures the purpose of the current group customer address.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can

predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import do not have an equivalent target object attribute, then review the Oracle Fusion Applications CRM Application Composer extensibility features for the group customer.

Target Import Objects Attributes Resources

To access the reference guide files for the group customer target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Group Customer File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
GroupProfile	<p>Includes detailed group customer information, such as, group name and group type. The default party usage for a group customer is SALES_PROSPECT.</p> <p>Sample attributes: AnnualIncome, NumOfDependents, MedianAge, and CertificationLevel.</p> <p>Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.</p>	HZ_IMP_PARTIES_T_Reference
Relationship	<p>Includes information about the relationship between the group customer and the group member. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</p> <p>Sample attributes: RelationshipCode and RelationshipType.</p> <p>Reference attributes: RelOrigSystem RelationshipID, and RelOrigSystemReference.</p>	HZ_IMP_RELSHIPS_T
Member	<p>Includes information about the group customer member.</p> <p>Sample attributes: ObjectKey and SubjectId.</p> <p>Reference attributes: RelOrigSystem RelationshipID, and RelOrigSystemReference.</p>	HZ_IMP_RELSHIPS_T

Classification	<p>Includes classification information for a group customer. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</p> <p>Sample attributes: ClassificationCode, ClassificationCategory, PrimaryFlag, and Rank.</p> <p>Reference Attributes: ClassificationOrigSystem and ClassificationOrigSystemReference.</p>	HZ_IMP_CLASSIFICS_T
AdditionalIdentifier	<p>Includes detailed information about an additional identifier for the group customer.</p> <p>Sample attributes: IdentifierValue, IdentifierType, Country, and IssuingAuthorityName.</p> <p>Reference attributes: IdentifierValue, AdditionalIdentifierOrigSystem, and AdditionalIdentifierOrigSystemRef</p>	HZ_IMP_ADDTNLPARTYID_T
AdditionalName	<p>Indicates alternative name of a group customer.</p> <p>Sample attributes: Additional Name, Additional Name Type, and GlobalPreferredNameFlag.</p> <p>Reference Attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemRefere</p>	HZ_IMP_ADDTNLPARTYNAMES_T
Url	<p>Indicates the Url of the group customer.</p> <p>Sample attributes: ContactPointType, PrimaryFlag, Url, and WebType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference</p>	HZ_IMP_CONTACTPTS_T_Reference

InstantMessenger	<p>Indicates the instant messenger or social networking information of a customer contact.</p> <p>Sample attributes: ContactPointPurpose, EndDate, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: CpOrigSystem, CpOrigSystemReference, and Objectkey.</p>	HZ_IMP_CONTACTPTS_T_Reference
Fax	<p>Indicates the fax of the group customer.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
FaxContactPreference	<p>Indicates the group customer preference about being contacted through fax.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
Mobile	<p>Indicates the mobile number of group customer.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference

MobileContactPreference	<p>Indicates the group customer's preference about being contacted through mobile phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
Phone	<p>Indicates the primary phone number of the group customer. If the group has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</p> <p>Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
PrimaryPhoneContactPreference	<p>Indicates the group customer's preference about being contacted through phone.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
Email	<p>Indicates the e-mail of the group customer.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and StartDate.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference

EmailContactPreference	<p>Indicates the group customer's preference about being contacted through e-mail.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
Address	<p>Indicates the address of a group customer. If the group has multiple addresses, one of the addresses is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, State, and PostalCode.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes or party sites: SiteOrigSystem, SiteOrigSystemReference, and PartysiteId1.</p>	HZ_IMP_LOCATIONS_T, HZ_IMP_PARTYSITES_T
AddressContactPreference	<p>Indicates the group customer's preference about being contacted at the primary address.</p> <p>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem, CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
AddressPurpose	<p>Indicates the purpose of a group customer's address.</p> <p>Sample attributes: SiteUseType, EndDate, and StartDate.</p> <p>Reference attributes: SiteOrigSystem and SiteOrigSystemReference.</p>	HZ_IMP_PARTYSITEUSES_T

Importing Legal Entities

Importing Legal Entities Using File-Based Import: Explained

This topic explains how to prepare and import legal entities from an external data source into Oracle Fusion Applications, using the File-Based Data Import feature.

A legal entity is a recognized party with rights and responsibilities that is established by legislation. You create 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. Within Oracle Fusion Applications CRM, a legal entity is a representation of the legal unit of a customer.

You must consider the following questions before importing legal entity information:

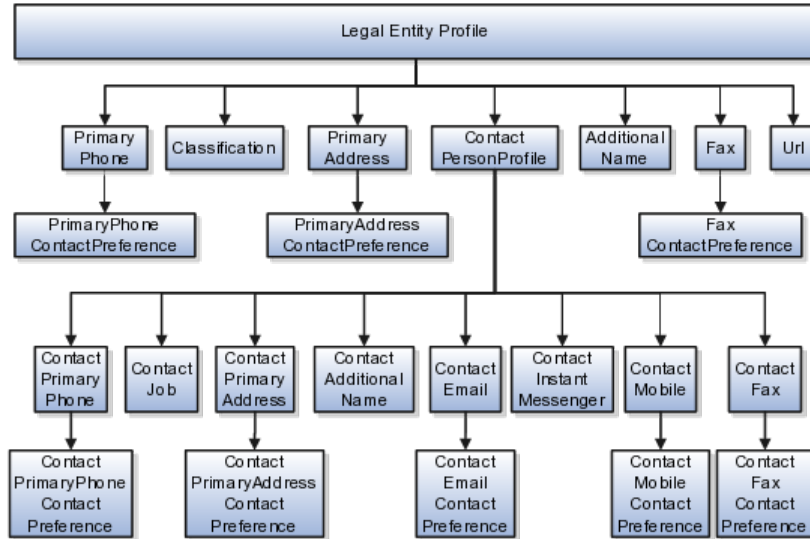
- How does your legacy or source system structure represent the legal entity information compared to how Oracle Fusion Applications represent the same data?
- How do your legacy or source system legal entities relate to business units, divisions, ledgers, balancing segments, consolidation rules, intercompany transactions, and payroll reporting?
- Do you have to configure values in Oracle Fusion Applications to map to your existing data values to the Legal Entity import object?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

Before you can import your legacy or source system legal entity data, you must first analyze the data and see how it corresponds with the legal entity object structure of Oracle Fusion Applications. You must understand how Oracle Fusion Applications represent the structure of the data for a legal entity.

The legal entity structure of Oracle Fusion Applications is hierarchical. The organization profile is at the root level of the legal entity. The organization profile must exist before you can import lower-level components, such as e-mail, classification, additional name, primary address, and fax. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the legal entity.

The following figure shows the legal entity and its child entities.



The organization profile contains basic information about the legal entity, such as the legal entity name and party usage. For each legal entity, you can assign classifications, contacts, phone details, and additional names. The contact of the legal entity, in turn, includes other child entities that capture information about the contact, such as contact job, contact primary phone, contact primary address, and contact e-mail.

Note

All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Comparing Business Object Data

After you understand the structure of the data, then compare the detailed attribute values of your data with the Oracle Fusion Applications data. Each import object is a collection of attributes organized to help you when mapping your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up a legal entity.

A good understanding of the import objects and attribute details is critical to preparing your import data. The attribute details of Oracle Fusion Applications are available for your reference in the Oracle Enterprise Repository. The reference guide files contain descriptions, default values, and validation for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correspond to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

If you have to extend the Oracle Fusion Applications import object to import your legacy or source data, you must use Oracle Fusion Applications CRM Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for

importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Legal Entities Using File-Based Data Import

For the legal entity business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. The Define File-Based Data Import Setup and Maintenance task list includes the tasks required to configure the import objects, create source file mappings, and schedule the import activities. You submit file-based import activities for each import object.

An import activity defines the instructions for processing the import data, including the source file, import mapping from the source file to the Oracle Fusion Applications object and attribute, and the import schedule.

When importing legal entity information, you first import the basic legal entity profile information and then the child entities for the legal entity. When importing child entities, you must provide the parent reference information for all parent levels for the entity. You must provide the PartyOrigSystem and PartyOrigSystemReference of the legal entity when importing contacts for the legal entity. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. For example, you first import basic profile details, such as first name, last name, party type, and party usage. You then import contacts and contact information, such as phone, address, contact points, and fax for the legal entity.

Note

When importing contact information for a legal entity, you must provide the relationship reference information. This information is required because a contact can have multiple relationships with a legal entity.

Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity. Alternatively, you can also navigate to the Define Legal Entities task list from the Setup and Maintenance work area to view the legal entities that you have imported.

Legal Entity Import Objects: How They Work Together

The Legal Entity import object allows you to import legal entities, their detailed information, and contacts related to the entity. This topic describes the Legal Entity import object. It introduces the following information:

- Target import object concepts
- Target objects for the Legal Entity import object
- Target import object attributes
- Target import object attribute reference guide files for evaluating and mapping source file data

Legal Entity Target Import Object Concepts

The Legal Entity import object imports legal entity information and its contacts. The Legal Entity import object is split into separate target import objects for organizing the individual attributes for the different aspects of the legal entity and its contact. Before you import data, you must understand how the target objects are related and what attributes are included in each object. This information helps you create the mapping between the source data in your import file and the target attributes in the application.

The target import objects in the Legal Entity import object are grouped into information about the legal entity and the contact. The organization profile is the target import object containing attributes to import information about the legal entity. You can have multiple contacts associated with a legal entity. There are multiple target import objects that include attributes to import contacts and their related information.

When updating an existing legal entity with more information, you must provide the parent reference information for the existing legal entity. When importing contacts or contact information for a legal entity, you must provide relationship reference information in addition to the parent reference. This information is required because a contact can have multiple relationships with the legal entity. When importing information about a contact, you must refer to the specific relationship you want to import information for. For example, you might want to import information about John Smith the employee or John Smith the board member. If you do not include the reference information for a contact relationship, then the import process creates a relationship.

To update the information for an existing legal entity or to create a legal entity, you can import legal entity profile information, addresses, and contact points, such as phone and fax. The following target import objects are for creating and updating the corresponding legal entity information: OrganizationProfile, PrimaryAddress, PrimaryPhone, Url, Fax, Classification, AdditionalName, and ContactPersonProfile.

To update or create a contact, use the following target import objects: ContactPersonProfile, ContactJob, ContactPrimaryAddress, ContactEmail, ContactPrimaryPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference files contain descriptions, logic used for default values, and validation information for each of the Oracle Fusion Applications attributes. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import

activity using File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import do not have an equivalent target object attribute, then review the Oracle Fusion Applications CRM Application Composer extensibility features for the marketing response.

Target Import Objects Attributes Resources

To access the reference guide files for the legal entity target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Legal Entity File-Based Data Import asset.

Target Import Object	Description	Attribute Reference File Name of Oracle Enterprise Repository
OrganizationProfile	Includes detailed legal entity information, such as, organization name and organization type. Sample attributes: OrganizationSize, RegistrationType, LegalStatus, and RegistrationType. Reference attributes: PartyOrigSystem, PartyOrigSystemReference, and PartyId.	HZ_IMP_PARTIES_T_Reference
PrimaryPhone	Indicates the primary phone number of the legal entity. If the legal entity has multiple phone numbers, one of the phone numbers is designated as the primary phone number. Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber. Reference attributes: CpOrigSystem and CpOrigSystemReference.	HZ_IMP_CONTACTPTS_T_Reference
Fax	Indicates the fax of the legal entity. Sample attributes: PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, and PhoneNumber. Reference attributes: CpOrigSystem and CpOrigSystemReference.	HZ_IMP_CONTACTPTS_T_Reference

Url	<p>Indicates the Url of the legal entity.</p> <p>Sample attributes: ContactPointType, PrimaryFlag, Url, and WebType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference</p>	HZ_IMP_CONTACTPTS_T_Reference
AdditionalName	<p>Indicates alternative name of a legal entity.</p> <p>Sample attributes: Additional Name, Additional Name Type, and GlobalPreferredNameFlag.</p> <p>Reference attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemRefere</p>	HZ_IMP_ADDTNLPARTYNAMES T_Reference
Classification	<p>Includes classification information for a legal entity. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</p> <p>Sample attributes: Classification code, Classification category, PrimaryFlag, and Rank.</p> <p>Reference attributes: ClassificationOrigSystem and ClassificationOrigSystemReference.</p>	HZ_IMP_CLASSIFICS_T_Reference
PrimaryAddress	<p>Indicates the primary address of a legal entity. If the contact has multiple addresses, one of the addresses is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, and State.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes for party sites: SiteOrigSystem, SiteOrigSystemReference, and PartysiteId1.</p>	HZ_IMP_PARTYSITES_T_Reference

ContactPersonProfile	<p>Includes detailed information, such as, person name, relationship type, and marital status, of a legal entity contact.</p> <p>Sample attributes: Gender, MaritalStatus, PartyType, PartyUsageCode, PersonFirstName, and PersonLastName.</p> <p>Reference attributes: PartyOrigSystem, PartyID, PartyOrigSystemReference, RelOrigSystem, RelOrigSystemReference, and RelationshipId.</p>	HZ_IMP_RELSHIPS_T_Reference
ContactEmail	<p>Indicates the e-mail of the legal entity contact.</p> <p>Sample attributes: ContactPointPurpose, EmailAddress, and start date.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactFax	<p>Indicates the fax of the legal entity contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, and PhoneExtension.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactInstantMessenger	<p>Indicates the instant messenger or social networking information of a legal entity contact.</p> <p>Sample attributes: ContactPointPurpose, end date, InstantMessagingAddress, and InstantMessengerType.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference

ContactJob	<p>Includes job information of a legal entity contact.</p> <p>Sample attributes: Comments, ContactNumber, Department, DepartmentCode, and JobTitleCode.</p> <p>Reference attributes: RelOrigSystem, RelationshipID, and RelOrigSystemReference.</p>	HZ_IMP_CONTACTS_T_Reference
ContactMobile	<p>Indicates the mobile number of a legal entity's contact.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactPrimaryAddress	<p>Indicates the primary address of a legal entity contact. If the contact has multiple addresses, one of the addresses is designated as the primary address.</p> <p>Sample attributes: Address1, Address2, City, Country, County, and State.</p> <p>Reference attributes for location: LocationOrigSystem, LocationOrigSystemReference, and LocationId1.</p> <p>Reference attributes for party sites: SiteOrigSystem, SiteOrigSystemReference, and PartysiteId1.</p>	HZ_IMP_PARTYSITES_T_Reference

ContactPrimaryPhone	<p>Indicates the primary phone number of the legal entity contact. If the legal entity has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</p> <p>Sample attributes: ContactPointPurpose, ContactPointType, PhoneAreaCode, PhoneCountryCode, PhoneExtension, PhoneLineType, PhoneNumber, and PrimaryFlag.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPTS_T_Reference
ContactAdditionalName	<p>Indicates alternative name of a legal entity contact.</p> <p>Sample attributes: NameId, PersonFirstName, PersonLastName, and PersonTitle.</p> <p>Reference attributes: AdditionalNameOrigSystem and AdditionalNameOrigSystemRefere</p>	HZ_IMP_ADDTNLPARTYNAMES_T_Reference
PrimaryPhoneContactPreference	<p>Indicates the legal entity's preference about being contacted through phone.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
MobileContactPreference	<p>Indicates the legal entity's preference about being contacted through mobile.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

FaxContactPreference	<p>Indicates the legal entity's preference about being contacted through fax.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
EmailContactPreference	<p>Indicates the legal entity's preference about being contacted through e-mail.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactPrimaryAddressContactPre	<p>Indicates the legal entity contact's preference about being contacted at the primary address.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
ContactPrimaryPhoneContactPrefe	<p>Indicates the legal entity contact's preference about being contacted through the primary phone number.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

PrimaryAddressContactPreference	<p>Indicates the legal entity's preference about being contacted at the primary address.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference
PrimaryPhoneContactPreference	<p>Indicates the legal entity's preference about being contacted through the primary phone number.</p> <p>Sample attributes: PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</p> <p>Reference attributes: CpOrigSystem and CpOrigSystemReference.</p>	HZ_IMP_CONTACTPREFS_Reference

Importing Resource Teams

Importing Resource Teams Using File-Based Import: Explained

This topic explains how to prepare and import resource team data from an external data source into Oracle Fusion Applications using the File-Based Data Import feature. A resource team is a temporary group of resources that is formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization. You enter your resource team information using the Setup and Maintenance work area, Manage Resource Teams task, or you can import data to create new or update the existing resource teams.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the resource team compared to how Oracle Fusion Applications represent the same data?
- Do you have to configure values in Oracle Fusion Applications to map to your data values?
- Do you have to customize Oracle Fusion Applications to capture additional attributes that are critical to the way you do business?

- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your resource team data corresponds with the data in Oracle Fusion Applications in order to be able to map your legacy data to the data needed by Oracle Fusion Applications. First, you must understand how Oracle Fusion Applications represent the structure of the data for a resource team.

In Oracle Fusion Applications, one table stores the resource team definition and other tables optionally store profile details for that resource team. Profile details for a resource include information about a resource team's organization and team memberships.

Import Objects for the Resource Team

To facilitate the import of resource teams, Oracle Fusion Applications incorporate the structure of the resource team into import objects. The import object for the resource team is Resource Team.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Fusion Applications data and to support one-to-many relationships between the structural components that make up the resource team.

A good understanding of the attribute details of the import objects is critical to preparing your import data. The reference guide files contain descriptions, logic used to choose default values, and validation information for each of the Oracle Fusion Applications attributes. The validation information includes the navigation to the task where you can define values in Oracle Fusion Applications. For example, if you have values in your data that correlate to a choice list in Oracle Fusion Applications, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

Import Object	Related Import Object Topic
Resource Team	Resource Team Import Objects: How They Work Together

Hint: You can use the keyword importing resource teams to search for related topics in Oracle Fusion Applications Help.

Extensible Attributes

If you need to extend the Oracle Fusion Applications object to import your legacy or source data, you must use Oracle Fusion CRM Application Composer to design your object model extensions and to generate the required artifacts

to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Resource Teams Using File-Based Data Import

For the resource team business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks needed to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When creating a new contact, you import the Resource Team object.

You must be assigned the Master Data Management Administrator job role to access and submit the import activities for resource teams.

Verifying Your Imported Data

Oracle Fusion Applications provide File-Based Import activity reports, which can be used to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Resource Teams work area to view the imported resource teams.

Resource Team Import Objects: How They Work Together

This topic describes the Resource Team import object. You use two main import objects, ResourceTeam and ResourceTeamUsage, when you submit a file-based import activity to import resource team information.

This topic introduces the following:

- Target import object concepts
- Target objects for the Resource Team import object
- Target import object attributes
- Target object attribute reference guide files

Target Import Object Concepts

The Resource Team import object is used to import resource teams and information to create or update associated organizations or persons. The Resource Team import object is split into separate target import objects for organizing the individual attributes of the different aspects of the resource team and the resource team's usage. To map the source data in your import file to the

target attributes in Oracle Fusion Applications, you must understand how the target objects are related and what attributes are included in each target object.

Response Target Import Objects

The target import objects in the Resource Team import object are generally grouped into information about the resource team and its usage. The ResourceTeam target import object contains information that identifies the resource team, such as the team's name, ID, active dates, and so on. The ResourceTeamUsage target import object contains information that captures the usage of the resource team.

To import or update a resource team, use the ResourceTeam target import object. To import or update a resource team's usage, use the ResourceTeamUsage target import object.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and their valid values. To evaluate your source data and Oracle Fusion Applications attributes for mapping and validation, you use an Oracle Enterprise Repository reference guide, which is available for each target import object. The reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the File-Based Import Mapping task, or you can define the mapping when you define the import activity using the File-Based Import Activity task. Both tasks are available in the Setup and Maintenance work area.

Note

If any of the attributes you want to import do not have an equivalent target object attribute, then review the Oracle Fusion CRM Application Composer extensibility features for the marketing response.

Target Import Objects Attributes Resources

To access the reference guide files for the resource team's target import objects, see the File-Based Data Import assets in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

The following table lists the reference guide files that are available from the Documentation tab for the Resource Team File-Based Data Import asset.

Target Import Object	Description	Reference Guide File Names
ResourceTeam	Captured information that identifies a resource team. Sample attributes: TeamId, TeamName, TeamDescription, and Language.	HZ_IMP_TEAMS_Reference

ResourceTeamUsage	<p>Contains information that indicates the usage of the resource team.</p> <p>Sample attributes: RequestId, TeamUsageId, TeamId, and Usage.</p>	HZ_IMP_TEAM_USAGES_Reference
-------------------	---	------------------------------

FAQs for Define File-Based Data Import

What determines the list of objects displayed?

A single import object can have multiple associated components that are considered objects by themselves. Whether or not an associated object can be grouped as a component of another object for the purpose of file import is determined by the complexity of the object structure and how it is stored in the data model. Oracle Fusion provides import objects predefined to meet the file processing import requirements. Consequently, in some cases, more than one source file may be required to capture all associated components of an object.

What happens if I inactivate an Import Activity?

The Import Activity will not stop the currently running process. However, it will stop the next process that has not started plus any future repeating file import activities. You can always activate the process at a later stage.

What happens if I add a marketing list in the Import Activity definition?

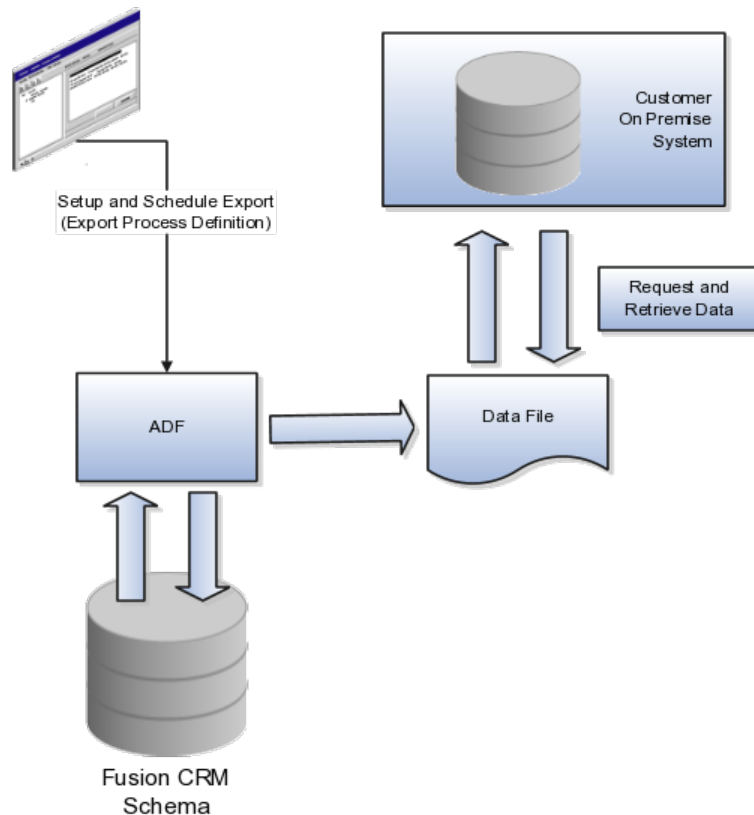
File-based data import enables you to record consumers and organization contacts in a marketing list when importing consumer, lead, and response import objects. Select an existing list or create a new one. A marketing list is assigned the list type value of Imported if created while defining an import activity. After the objects are imported successfully, the consumers and contacts are added as members of the marketing list.

Common Customer Data Management Configuration: Manage Bulk Data Export

Bulk Export: Overview

The Bulk Export application provides a mechanism to extract large volumes of data from Fusion CRM objects. These extracts can be the full set of records for an object or incremental extracts. For example, data extracted for a specific period of time, from the hosted CRM system to an on-premise database that resides behind a user's fire-wall. The system will create comma separated variable or tab delimited files with the extracted data, which will be available to users as attachments to the batch records that have been executed.

The following figure depicts the process of selecting data for export, scheduling and finally delivering the exported data file.



Bulk Export Process Definition: Explained

This solution provides a mechanism to extract large volumes of data from Fusion CRM objects, both as extracts of a full set of records for an object as well as incremental extracts. The system will create comma or tab delimited files with the extracted data which will be available to users as attachments to the batch records that have been executed.

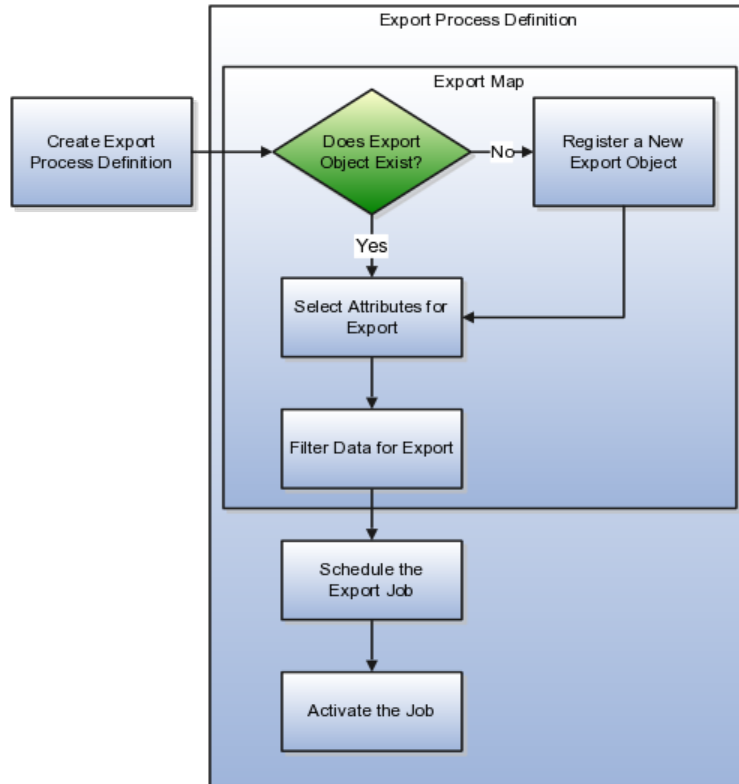
In order to create the extracts, two steps must be completed. First, mapping files for the full and incremental extract processes must be defined in the Fusion CRM system. These maps will specify which columns and filters will be applied to each export process for each export object. For the incremental extracts, filters can be created that leverage time stamps to determine which rows will be queried out of the system. All mapping files will be saved in the system and reused for each extract.

Next, the hourly and weekly data export processes are scheduled in the Fusion export tool. For any required incremental and scheduled export, the export task should either exist or be created through the UI. Oracle Web Services would only be used to schedule the export and start it. After each export process executes and completes, a comma or tab delimited data file will be created and stored in the Fusion system as an attachment. The formatted file can be downloaded by using the `getAttachment()` web service or by using the interactive UI in the export tool.

There are no transactional steps for this process in the Fusion CRM application, there are only prerequisite setup steps. Once these steps are complete the process

should run automatically. The prerequisite steps in Fusion are to create an export map and export job schedule for each object to be extracted (this only needs to be done once).

The Bulk Export Process Definition is made up of the Export Map and the processing schedule. See the steps below.



Select the Export Object

The export object is the Fusion data base object where the data resides. It is made up of attributes. If you need to export data from a custom table, you must register the object as an export object. This is accomplished from the Manage Export Process UI, Manage Export Objects action. All the delivered tables and their attributes are available for export.

Select Attributes

The export object is made up of attributes. These attribute may be selected for export or not included. You can edit the header text of the attribute to make its meaning more clear to other users of this process.

Filter Criteria

Each attribute may have limits or conditions enforced. Various operators are available for selecting the data to precisely select the data required for the export. You can save the filter criteria and then modify the criteria and save it under a new name. You can then change the filter by coming here to select an alternate filter name. Because the filters are related to the export object, if you reuse a map and change the filter, you are changing it for any Export Process Definition that

uses that map. The attributes you use for the map have no bearing on what is available in the filter. All fields from the VO are available for use in the filter. For example, you can filter by TYPE but not show TYPE in the output.

Schedule the Export Process

Once defined, the export process is scheduled. You can run the process immediately or at the time and date of your choosing. If you decide to schedule the job at a later date you can also choose to set up a recurring schedule of extracts.

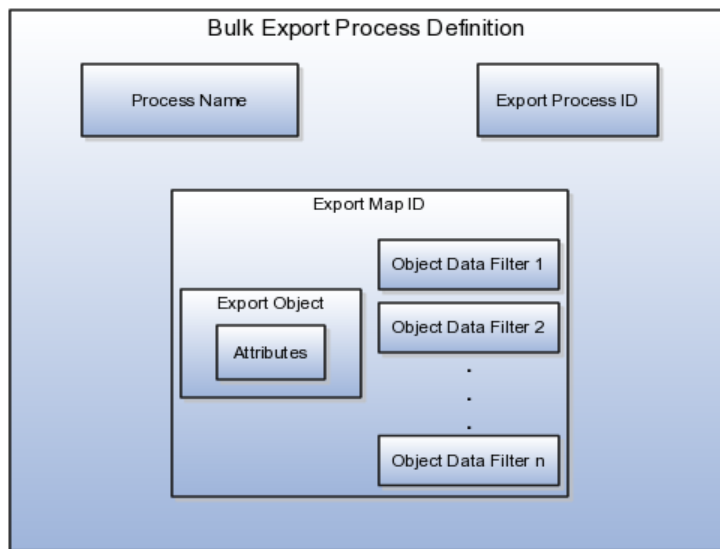
Activate the Job

By clicking on the Activate button, you make the job available to be run. It does not start an export process.

Bulk Export Process Components: How They Work Together

In the two step process used by Oracle Fusion Bulk Export, the first is the mapping of files for the full and incremental extract processes. The second step is the scheduling of the export. You create a process definition that includes both of these steps.

The process definition has three components that together make exporting data easier by leveraging the export maps that you have already built. The process name, the export process ID and the export map ID all serve to identify the specific process definition as well as leverage your work with reusable export maps



Process Name

A user-supplied, natural language way to refer to the Export Process Definition. This enables you to refer to the export process definition easily rather than using the machine generated ID. For example, use Customer or some other meaningful name as the export process name instead of the export process ID 100000019897192.

Export Process ID

A unique, system generated identifier for the export process definition that ties together the export map, with its export objects and filters, and the defined export schedule.

Export Map ID

A unique identifier for the export map itself. You can name the export map or leave the field blank for a system generated map name to be entered. You can reuse the export map in different process definitions. For example, you could create a process definition to export all the data from the Customer export object. You could then reuse that export map and apply a new filter on the data to create an incremental export, such as data accrued since the last export date.

FAQs for Manage Bulk Data Export

How can I tell which objects to select?

Review the requirements for the data to be exported and determine the source view object that holds the attributes you want.

How can I create a subset of data for export?

Full sets of data are not always required for export. To create a subset of data, use filter criteria to determine the time frame or scope of data, based on values of the attributes. For example, to find activities for a certain period, use a project start date from 1/1/11 through 3/31/11, navigate to the Export Objects Detail Sub Page and click the filter icon. Fill in the filter criteria dialog for the project start dates to select the data to be exported. You run the export by navigating to the **Setup and Maintenance** menu, selecting **Manage Task Lists and Tasks**. Then, search for **Schedule Export Processes** and click the **Go to Task** icon on the line for this task.

How can I see my exported data?

You can look on the Schedule Export Processes, Overview page to see the History subpage. The column Exported Data File shows a hyperlink to your output file. This file will be a comma separated variable or a tab delimited file. Click that link to open the file and see the exported data.

How can I use my own View Object for export?

You can use your own defined view objects as a source for Bulk Export. To register your view objects for export, select **Setup and Maintenance** from the

Tools menu and search for the **Manage Export Objects** task. Click the **Go to Task** icon and on the **Manage Export Objects** page click the **Create** icon to add your **View Object**, making it available for use.

What happens if I change the sequence number or header text?

Changing the sequence number changes the order of the attributes in the exported data file. Changing the header text enables you to give a more intuitive meaning to the attribute and the associated data.

What happens if I need data from more than one view object?

Select as many view objects as required to be export objects for the export process. Choose the individual attributes required from each export object.

Define Data Quality

Define Data Quality: Explained

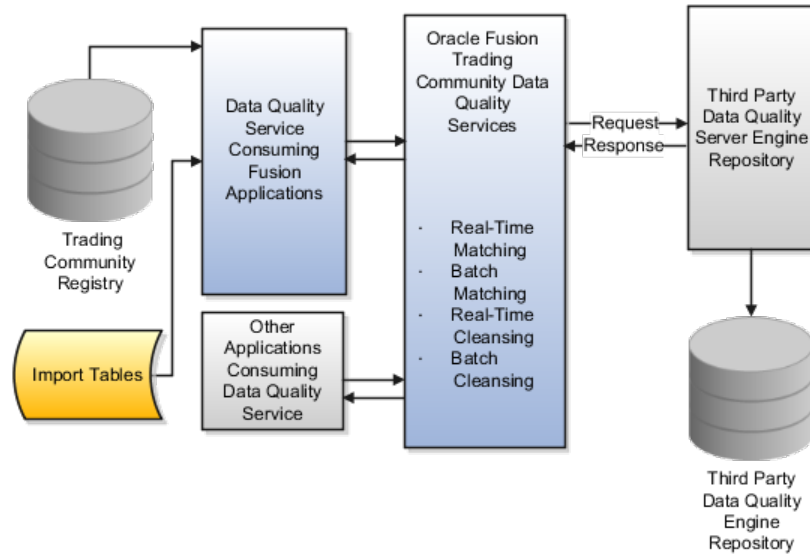
Defining data quality involves setting up two functionalities, matching and address cleansing, by performing related regular and manual setup and maintenance tasks. Manual setup and maintenance tasks are performed on the related third-party data quality (DQ) engine, Informatica Identity Resolution (IIR).

The Oracle Fusion Trading Community Data Quality matching and address cleansing services are designed to cater to all Oracle Fusion applications that use the trading community registry.

Matching is the identification of potential duplicates for organizations, persons, and locations. Potential duplicate records are identified and displayed in the data quality functionality consuming Oracle Fusion Applications during a variety of activities such as duplicate prevention in real-time when a customer record is being created or duplicate identification in batch mode for existing records.

Address cleansing is the process of correction and validation of address data, based on postal requirements. For example, if a user enters valid values for postal code, city, and country, the data quality functionality may return a value for state. Likewise, if a user enters valid values for city, state, and country, the data quality functionality automatically supplies a postal code value.

This figure illustrates the data quality management architecture. The data quality services receive matching or cleansing requests, such as duplicate prevention or address validation, from a variety of consuming applications, such as Oracle Fusion Sales and Oracle Fusion Trading Community Hub, and then internally invoke the appropriate third party services based either on a default configuration or a specific configuration passed to the services.



Matching and address cleansing are independent functionalities and are set up separately using the following steps:

- Setting up the third-party data quality (DQ) engine, Informatica Identity Resolution (IIR)

Note

You can find the IIR vendor documentation in <PROV.TOP>/InformaticaIR/pdf directory within the IIR Installation.

- Setting up the Fusion DQ Connector to IIR, by performing data quality related regular setup and maintenance tasks, such as defining and managing lookup choices and configurations used in data matching, cleansing, and matching index synchronization
- Connecting Oracle Fusion Applications to the third-party data quality (DQ) engine, Informatica Identity Resolution (IIR)

Managing Lookups

Review and define lookup values that provide choices for data matching and cleansing, such as address fields, quality error messages, and match tolerances.

Managing Data Quality Configurations

Perform the configurations required to enable data quality processes such as data quality matching and cleansing operations.

You can enable data quality processes such as matching and address cleansing for real-time or batch matching and cleansing, and as part of the data import process.

Real-time matching can prevent individual entry of duplicate trading community entities, such as organization, person, or location, into the trading community registry. Batch matching leads to identification of duplicates entries in the trading community registry.

As part of defining and managing data quality configurations, you search, review, and edit seeded matching and cleansing configurations for real-time and batch matching and cleansing of trading community entities.

Managing Server Configurations

Define, review, and update the data quality server configurations to integrate with the embedded data quality engine.

Next, associate the server name with the matching and cleansing configurations to perform duplicate prevention, batch duplicate identification, and real-time and batch data cleansing.

Managing Matching Index Synchronization

Review and update matching index synchronization options, to synchronize Oracle Fusion Trading Community Hub registry data with the data quality engine repository, such as system and identity tables. View the latest synchronization results, evaluate errors, and reset for continued processing after resolving error conditions.

Manage Data Quality Engine

Informatica Identity Resolution Setup: Procedures

Follow this procedure to verify the installation of the third-party data quality (DQ) engine, Informatica Identity Resolution (IIR), used by Oracle Fusion Trading Community Data Quality, and perform IIR setup operations. The IIR setup operation are manual setup and maintenance tasks performed using the UI of the third-party data quality (DQ) engine.

Note

If you are working on the 64-bit operating systems of Oracle Enterprise Linux, Windows, AIX, or Solaris Sparc, when you provision CRM using the provisioning framework, an instance of IIR (64 Bit) is automatically provisioned along side. However, for Solaris X-64 Bit platform, IIR does not get provisioned. You need to install it manually on any of the other 64-Bit operating systems. Except for the manual installation of IIR on a different platform, all the setup steps for Solaris X-64 Bit platform are identical to other platforms.

Performing Post-Provisioning Checks

Once Fusion Application environment is provisioned by Oracle Fusion Application Provisioning Framework, do the following to verify IIR installation:

- Check whether InformaticaIR directory exists under `PROV.TOP`.
 - Check whether IIR is already running using `ps -ef | grep InformaticaIR`. There should be eight processes (or four pairs of parent-child processes) running per server.
-

Note

If IIR post-provisioning verifications fail, perform IIR maintenance activities.

This table lists the processes that run when IIR is working.

Server Name	Process Name	Port	Purpose
License Server	ssalisp	1601	Meant for licensing matching requests. You can configure the value of this port.

Search Server	ssasrsv	1666	Serves an incoming matching request. Connects to the Rulebase server for system or Rulebase information.
Connection Server	ssacssv	1667	Meant for improving performance of search clients.
Console Server	ssacosv	1669	Enables Informatica IR Console Client to access the Rulebase and other files and to launch other IIR utilities.
Rulebase Server	ssasrsv	1668	Meant for accessing the IIR Rulebase in the database and is relevant only for matching. There is no separate process for this. It is started as part of ssasrsv.

An instance of IIR uses seven ports. If IIR is provisioned through the framework, the value of the license server port is determined by the value of the property `provisioning.setup.app-ports.core.iir.port` that is set in the provisioning plan. If IIR is installed manually, the value of the license server port is determined by `IIR_INSTANCE_1_PORT` in the `install.props` file.

The seven ports are listed here. For example, if the port value set for `provisioning.setup.app-ports.core.iir.port` uses the default value 1601, the ports occupied for Informatica Identity Resolution would have the following values:

- `SSA_LIPORT`= <value set for the property `provisioning.setup.app-ports.core.iir.port`>=1601
- `SSA_SEPORT`=1666
- `SSA_COPORT`=1667
- `SSA_RBPORT`=1668
- `SSA_CSPORT`=1669
- `SSA_XMPORT`=1670
- `SSA_HTTPORT`= <value set for the property `provisioning.setup.app-ports.core.iir.port`> + 11= 1612

Note

The value of `SSA_LIPORT` can be found in `<PROV.TOP>/InformaticaIR/env/envs` and the values of all other ports can be found in `<PROV.TOP>/InformaticaIR/env/iss`.

Post-Provisioning Operations

When post-provisioning checks are successful, perform these post-provisioning operations sequentially to set up IIR:

1. IIR Matching Setup. For more information, see: Informatica Identity Resolution Matching Setup: Procedures.

2. IIR Address Cleansing Setup. For more information, see: Informatica Identity Resolution Address Cleansing Setup: Procedures.
3. Oracle Fusion Data Quality Connector Setup. For more information, see: Fusion Data Quality Connector Setup: Procedures.

Setting up Multiple Informatica Identity Resolution Instances: Overview

Set up multiple Informatica Identity Resolution (IIR) instances to make use of client load balancing capabilities of IIR to implement failover. You can configure the Oracle Fusion Data Quality Connector to allow Oracle Fusion Applications to speed up data quality management operations and to avoid overload by sending matching and cleansing calls to different IIR instances. Note that the multiple instances do not require any data or repository replication and can work with one Oracle Fusion Application Database.

Note

The setup mentioned in this section is not mandatory and should be used only based on use cases, and in a specific deployment scenario.

Use multiple instances of the IIR and setup the Oracle Fusion Data Quality Connector appropriately to make use of load balancing to implement failover as follows:

- Load Balancing by matching and cleansing on different dedicated hosts
- Load balancing by matching with a secondary match server
- Load balancing and failover by matching and cleansing using a secondary instance

For more information on these load balancing and failover scenarios, refer to the topics in the Related Topics section of this topic.

Note

Informatica Identity Resolution (IIR) Server does not support native server load balancing.

Assumptions and Constraints

Take note of the following assumptions and constraints before using multiple instances of the IIR and the Oracle Fusion Data Quality Connector setup for load balancing and failover:

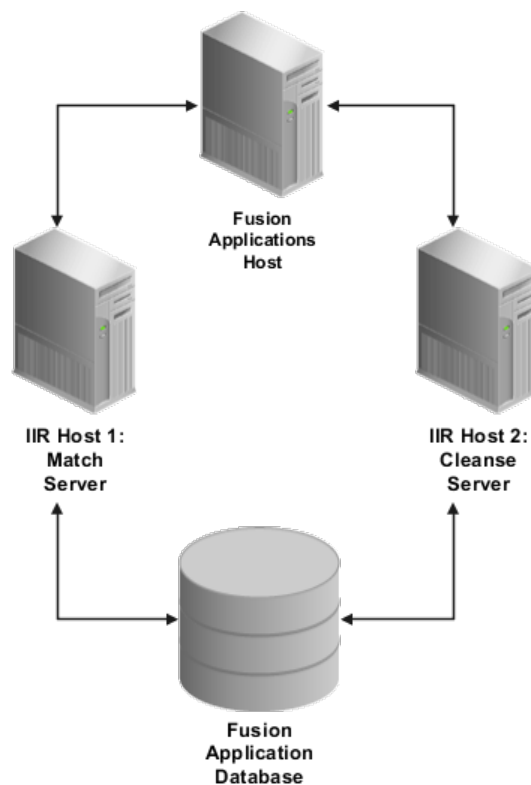
- CRM is provisioned with only one instance of IIR. Consequently, load balancing and failover using multiple instances of the IIR and the Oracle Fusion Data Quality Connector setup require manual installation of IIR for every additional instance. Since the installation of IIR through provisioning includes a bare minimum of IIR schema objects required in Oracle Fusion Applications Database, make sure that those schema objects are not installed again during the manual installation of IIR.
- Load balancing cleansing is not described as a separate scenario. Instead, the steps for load balancing and failover of cleansing operations are included as optional steps in the scenarios for load balancing and failover of matching operations.
- One or more instances of Oracle Fusion Applications and IIR may be installed on an Oracle Fusion Applications host.

- You cannot load balance and failover the update synchronizer. It can run only on one host.
- Any rollback of changes, for example reverting to a single instance, while load balancing using a secondary server requires a `delete_secondary_server.sql` and also requires the bouncing of application servers hosting Oracle Fusion Applications that use real-time data quality.

Load Balancing by Matching and Cleansing on Different Dedicated Hosts: Procedures

Use this procedure for load balancing with no specific requirement for high availability, or failover, by performing the matching and cleansing operations on dedicated hosts.

This figure provides a visual representation of the load balancing process achieved by matching and cleansing on different dedicated hosts.



Configuring the primary and Secondary IIR Hosts

Use the following steps to configure the IIR hosts:

1. Manually install Informatica Identity Resolution (IIR) on the secondary host, making sure `INSTALL_IIR_OBJECTS=N`. In other words, make sure that the schema objects installed during installation of IIR through

provisioning are not installed again during the manual installation of IIR. For more information, see: [Manually Installing Informatica Identity Resolution: Procedures](#)

2. Perform all the matching setup steps in the primary IIR host.
3. Perform all the cleansing setup steps in the secondary IIR host.

Configuring the Oracle Fusion Data Quality Connector

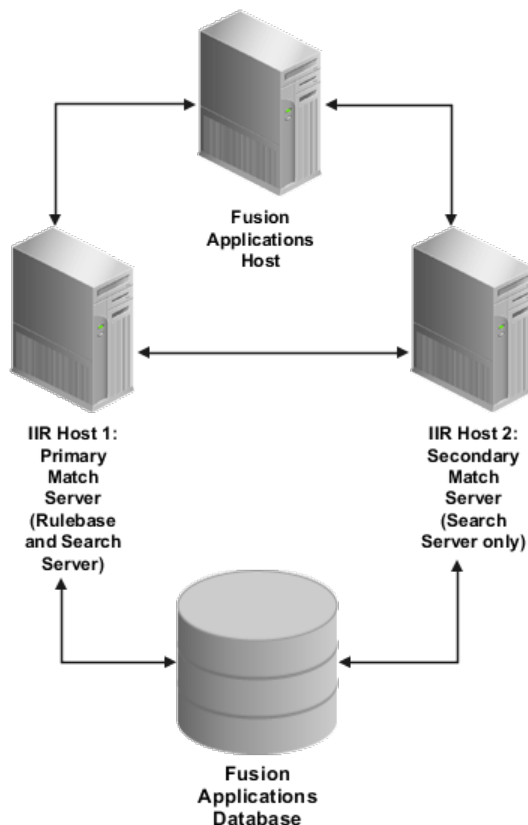
Use the following steps to configure Oracle Fusion Data Quality Connector:

1. In the Real-time and Batch Basic Server, enter the server address and server port of the primary IIR host, the IIR Matching Server.
2. In the Real-Time Cleanse Server and Batch Cleanse Server, enter the server address and server port of the secondary IIR host, the IIR cleansing server.

Load Balancing by Matching and Cleansing with Secondary Servers: Procedures

Use this procedure for load balancing with no specific requirement for high availability, or failover, by performing the matching operations on a secondary match server.

This figure provides a visual representation of the load balancing process for matching operations achieved by matching with a secondary match server.



Configuring the Secondary Match Server

Use the following steps to configure the secondary match server:

1. Manually install Informatica Identity Resolution (IIR) on the secondary match server, making sure `INSTALL_IIR_OBJECTS=N`. In other words, make sure that the schema objects installed during installation of IIR through provisioning are not installed again during the manual installation of IIR. For more information, see: Manually Installing Informatica Identity Resolution: Procedures.
2. Make sure IIR on the secondary match server has access to an Oracle DB client to connect to the Fusion Application Database and the `tnsnames.ora` has all the relevant SID information. Shut down IIR on the secondary match server.
3. Perform all the matching setup steps using IIR on primary match server. Shut down IIR on primary match server.
4. Start IIR on primary match server.
5. Start IIR as a search server only on the secondary match server by doing the following:

```
<PROV.TOP>/InformaticaIR/bin > bash
<PROV.TOP>/InformaticaIR/bin > . ./setfusionEnv.sh
<PROV.TOP>/InformaticaIR/bin > . ./env/iss
<PROV.TOP>/InformaticaIR/bin > ./ssasrsv -n{SEPORT2} -h{RBHOST1}:{RBPORT1}
-1$SSATOP/iirlog/idssrsv.log -2$SSATOP/iirlog/idssrsv.err -3$SSATOP/
iirlog/ids.dbg &
```

Note

The Search Server in Host 2 is started by pointing its port (SEPORT2) to the Rulebase Server's host (RBHOST1) and port (RBPORT1) in the primary match server. Here:

- SEPORT2 = Search Server Port (SSA_SEPORT defined in InformaticaIR/env/iss on IIR Host 2. The default port is 1666).
 - RBHOST1 = Hostname or IP of IIR Host 1
 - RBPORT1 = Rulebase Server Port (SSA_RBPORT defined in InformaticaIR/env/iss on IIR Host 2. The default port is 1668).
-

6. On Host 1 (Primary Host), start the IIR Update Synchronizer process using the following steps:
 - a. Sign in the Informatica Identity Resolution (IIR) host machine.
 - b. Enter `cd <PROV.TOP>/InformaticaIR/bin`
 - c. Enter `setfusionEnv.sh`
 - d. Start the IIR console client using the admin option, `./idsconc -a`
 - e. Select **Run Synchronizer** on the **Tools** menu to launch the synchronizer.
 - f. In the Update Synchronizer dialog, select **All** as the value for **IDT Name**, use the default values for the rest of the fields, and click **OK**.

- g. Verify that the updated and newly created person records are available in IIR, by searching for persons in the Per-dup tab of IIR Web Search Client.
 - h. Sign out of the Informatica Identity Resolution (IIR) user interface.
7. In the Oracle Fusion DQ Connector, on the Manage Server Configuration page, update the `SecondaryServer1Address` and `SecondaryServer1Port` values respectively for the host name and port of the Secondary Server (Host 2). These parameters can be found in the Server Parameter Values section.
 8. Select the **Enable High Availability** checkbox for each server operation.
 9. Click **Save and Close**.

Note

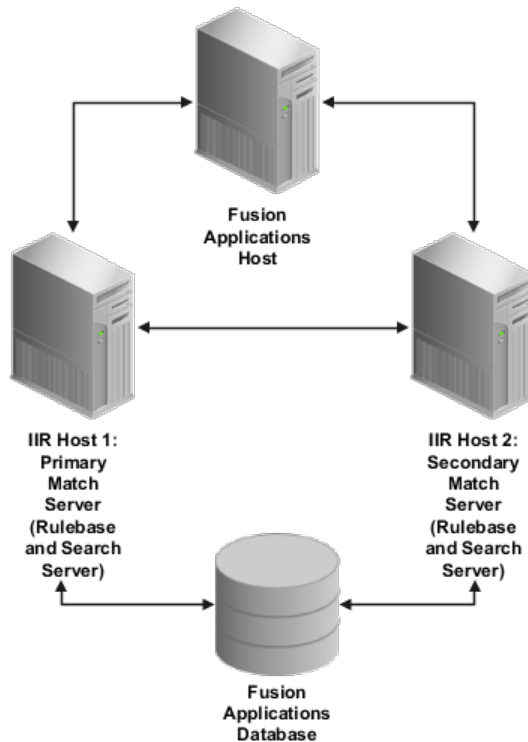
Repeat steps 7, 8, and 9 for each server configuration: Realtime and BatchBasic Server, Batch Cleanse Server, and Realtime Cleanse Server

-
10. Restart the Oracle Fusion Applications instances that require data quality management to enable Oracle Fusion Data Quality Connector to commence load balancing and failover.

Load Balancing and Failover by Matching and Cleansing Using a Secondary Instance: Procedures

Use this procedure for load balancing to implement high-availability, or failover, by performing both matching and cleansing operations using a secondary instance of Informatica Identity Resolution (IIR).

This figure provides a visual representation of the load balancing and failover process achieved by matching and cleansing using a secondary instance.



Configuring the Secondary instance of IIR to Load Balance and Failover

Use the following steps to configure the secondary instance of Informatica Identity Resolution (IIR) to load balance and failover matching and cleansing:

1. Manually install Informatica Identity Resolution (IIR) on the secondary host, making sure `INSTALL_IIR_OBJECTS=N`. In other words, make sure that the schema objects installed during installation of IIR through provisioning are not installed again during the manual installation of IIR. For more information, see: [Manually Installing Informatica Identity Resolution: Procedures](#).

Note

Since there are two fusion application instances in this scenario, the secondary IIR instance may already be installed. In such cases, you need to only configure the Oracle Fusion Data Quality Connector so that it points to the same Oracle Fusion Database as the primary IIR instance. For more information, see: [Oracle Fusion Data Quality Connector Setup: Procedures](#).

2. Make sure IIR on the secondary host has access to an Oracle DB client to connect to the Fusion Application Database and the `tnsnames.ora` has all the relevant SID information. Shut down IIR on the secondary host.
3. Perform all the matching setup steps using IIR on primary host. Shut down IIR on primary host.

4. Start IIR on primary host, Host 1, by starting the Rulebase Server first followed by starting the Search Server.

```
<PROV.TOP>/InformaticaIR/bin > bash
<PROV.TOP>/InformaticaIR/bin > . ./setfusionEnv.sh
<PROV.TOP>/InformaticaIR/bin > . ../env/iss
<PROV.TOP>/InformaticaIR/bin > ./ssasrsv -m{RBPORT1} -
gFusionRBSG,ssa:rtunitrb -wl -1$SSATOP/iirlog/priRB.log -2$SSATOP/iirlog/
priRB.err -3$SSATOP/iirlog/priRB.dbg &
<PROV.TOP>/InformaticaIR/bin > ./ssasrsv -n{SEPORT1} -
gFusionRBSG,ssa:rtunitrb -1$SSATOP/iirlog/priSE.log -2$SSATOP/iirlog/
priSE.err -3$SSATOP/iirlog/priSE.dbg &
```

5. Start IIR on the secondary host, Host 2, by starting the Rulebase Server first, followed by starting the Search Server.

```
<PROV.TOP>/InformaticaIR/bin > bash
<PROV.TOP>/InformaticaIR/bin > . ./setfusionEnv.sh
<PROV.TOP>/InformaticaIR/bin > . ../env/iss
<PROV.TOP>/InformaticaIR/bin > ./ssasrsv -m{RBPORT2} -
gFusionRBSG,ssa:rtunitrb -wl -1$SSATOP/iirlog/secRB.log -2$SSATOP/iirlog/
secRB.err -3$SSATOP/iirlog/secRB.dbg &
<PROV.TOP>/InformaticaIR/bin > ./ssasrsv -n{SEPORT2} -
gFusionRBSG,ssa:rtunitrb -1$SSATOP/iirlog/secSE.log -2$SSATOP/iirlog/
secSE.err -3$SSATOP/iirlog/secSE.dbg &
```

6. Start IIR Console on the primary host, Host 1.

```
<PROV.TOP>/InformaticaIR/bin > ./ssacssv -o -n{CSPORT1} -hsellocalhost:
{SEPORT1} -hcollocalhost:{COPORT1}
-hhtlocalhost:{HTPORT1} -gFusionRBSG,ssa:rtunitrb -w$SSATOP/ids
-1$SSATOP/iirlog/cons.log -2/$SSATOP/iirlog/cons.err
-3$SSATOP/iirlog/cons.dbg &
```

For example:

```
<PROV.TOP>/InformaticaIR/bin > ./ssacssv -o -n1669 -hsellocalhost:1666 -
hcollocalhost:1667
-hhtlocalhost:1612 -gFusionRBSG,ssa:rtunitrb -w$SSATOP/ids
-1$SSATOP/iirlog/cons.log -2/$SSATOP/iirlog/cons.err
-3$SSATOP/iirlog/cons.dbg &
```

7. Start IIR Connection Server on the primary host, Host 1.

```
<PROV.TOP>/InformaticaIR/bin > ./ssacosv -hlocalhost:{SEPORT1} -n{COPORT1}
-t5 -1$SSATOP/iirlog/conn.log
-2/$SSATOP/iirlog/conn.err -3$SSATOP/iirlog/conn.dbg &
```

For example:

```
<PROV.TOP>/InformaticaIR/bin > ./ssacosv -hlocalhost:1666 -n1667 -t5
-1$SSATOP/iirlog/conn.log
-2/$SSATOP/iirlog/conn.err -3$SSATOP/iirlog/conn.dbg &
```

Note

- RBPOR1 and RBPOR2 respectively are the ports for the Rulebase Server on IIR Host 1 and IIR Host 2 respectively. The values for these parameters should respectively be the value of SSA_RBPOR1 in `InformaticaIR/env/iss` on Hosts 1 and 2.
- SEPOR1 and SEPOR2 respectively are the ports for the Search Server on IIR Host 1 and IIR Host 2. The values for these parameters should respectively be the value of SSA_SEPOR1 in `InformaticaIR/env/iss` on Hosts 1 and 2.
- CSPOR1 is the port for the Console Server on IIR Host 1. The values for this parameter should be the value of SSA_CSPOR1 in `InformaticaIR/env/iss` on Hosts 1.
- COPOR1 is the port for the Connection Server on IIR Host 1. The values for this parameter should be the value of SSA_COPOR1 in `InformaticaIR/env/iss` on Hosts 1.
- HTPOR1 is the port for the HTTP Search Server on IIR Host 1. The values for this parameter should be the value of SSA_HTPOR1 in `InformaticaIR/env/iss` on Hosts 1.
- The parameter `w1` specifies that the polling frequency of the Secondary Server to determine if the Primary Server is active is one second. For example, if the value of the parameter is set to `w10` then the polling frequency of the Secondary Server to determine if the Primary Server is active will be 10 seconds.
- The steps 4 and 5 essentially start the Rulebase Servers in a group, making sure, that even if one matching instance fails, the other can continue to function. Only one rulebase server (primary) in the group is permitted to respond to queries and the other (secondary) server periodically polls the primary server to determine if it is active.
- In the scenario in which load balancing is done by matching with a secondary match server, if the primary server (Host 1) fails, matching will stop functioning unless it is restored.

In contrast, in the present scenario, if the primary server (Host 1) fails, the secondary server (Host 2) takes the role of the primary server, with no expectation of Host 1 to be restored. However, there is a small latency in the secondary server assuming the role of the primary server and there could be a loss of the matching functionality during this temporary latency period.

-
8. On Host 1 (Primary Host), start the IIR Update Synchronizer process using the following steps:
 - a. Sign in the Informatica Identity Resolution (IIR) host machine.
 - b. Enter `cd <PROV.TOP>/InformaticaIR/bin`
 - c. Enter `setfusionEnv.sh`
 - d. Start the IIR console client using the admin option, `./idsconc -a`

- e. Select **Run Synchronizer** on the **Tools** menu to launch the synchronizer.
 - f. In the Update Synchronizer dialog, select **All** as the value for **IDT Name**, use the default values for the rest of the fields, and click **OK**.
 - g. Verify that the updated and newly created person records are available in IIR, by searching for persons in the Per-dup tab of IIR Web Search Client.
 - h. Sign out of the Informatica Identity Resolution (IIR) user interface.
9. In the Oracle Fusion DQ Connector, on the Manage Server Configuration page, update the `SecondaryServer1Address` and `SecondaryServer1Port` values respectively for the host name and port of the Secondary Server (Host 2). These parameters can be found in the Server Parameter Values section.
 10. Select the **Enable High Availability** checkbox for each server configuration.
 11. Click Save and Close.

Note

Repeat steps 9 to 11 for each server operation: Realtime and BatchBasic Server, Batch Cleanse Server, and Realtime Cleanse Server.

12. Restart the Oracle Fusion Applications instances that require data quality management to enable Oracle Fusion Data Quality Connector to commence load balancing and failover.

Informatica Identity Resolution Server Maintenance and Administration: Procedures

Administration and maintenance includes start up and shutdown activities of the Informatica Identity Resolution (IIR) server and how Oracle database interacts with the IIR server. It also includes locking and unlocking of the Rulebase and how you can use the Informatica IR Console Client.

Starting Up and Shutting Down IIR

To start the IIR server, enter the following commands:

```
<PROV.TOP>/InformaticaIR/bin> bash
<PROV.TOP>/InformaticaIR/bin> ./setfusionEnv.sh
```

Note

If you are using C Shell, use the command `source setfusionEnv.csh`

```
<PROV.TOP>/InformaticaIR/bin> ./liup
<PROV.TOP>/InformaticaIR/bin> ./idsup
<PROV.TOP>/InformaticaIR/bin> ./idsconc -a
```

Note

Start the Update Synchronizer from the IIR Admin Console. However, after starting the Update Synchronizer, shut down the IIR Admin Console. For more information on Starting Up and Shutting Down Update Synchronizer, refer to the topic Informatica Identity Resolution Matching Setup: Procedures.

To shutdown the server, enter the following commands:

Note

If the Update Synchronizer is running (check for the `updsync` process), stop it from the IIR Admin Console. After stopping the Update Synchronizer, shut down the IIR Admin Console. For more information on Starting Up and Shutting Down Update Synchronizer, refer to the topic Informatica Identity Resolution Matching Setup: Procedures.

```
<PROV.TOP>/InformaticaIR/bin> bash
<PROV.TOP>/InformaticaIR/bin> ./setfusionEnv.sh
```

Note

If you are using C Shell, use the command `source setfusionEnv.csh`

```
<PROV.TOP>/InformaticaIR/bin> ./idsdown
<PROV.TOP>/InformaticaIR/bin> ./lidown
```

Note

It is strongly recommended that you verify the Informatica Identity Resolution after every startup or a restart.

Shutting Down Servers for High Availability

To shutdown the primary and secondary servers make sure the following steps are followed:

1. Stop Update Synchronizer on the primary server.
2. Stop license server on the Primary and all other servers on both Primary and secondary server as follows:

```
<PROV.TOP>/InformaticaIR/bin > bash
<PROV.TOP>/InformaticaIR/bin > . ./setfusionEnv.sh
<PROV.TOP>/InformaticaIR/bin > lidown
<PROV.TOP>/InformaticaIR/bin > ./ssashut -h<HOST>:<PORT>
```

Note

Do the `ssashut` for all servers started using their respective host and port.

Verifying Informatica Identity Resolution (IIR) Configuration

Before handing over an environment for business use, you need to perform certain checks to determine if IIR is in working condition. Perform the checks after performing any of following activities:

- Setting Up Informatica Identity Resolution sever for the first time
- Restarting the Informatica Identity Resolution sever for any reason such as after a maintenance window or a scheduled downtime (patching, bouncing the database, troubleshooting and so on)

If any of the IIR configuration checks fail, the data quality functionality will not work. For information on troubleshooting, refer to the Troubleshooting Informatica Identity Resolution and Data Quality Setup: Overview help topic listed in the Related Topics section.

Perform the following checks to verify Informatica Identity Resolution (IIR) configuration:

- IIR Server Check: Verify that the total number of processes running for IIR search components is eight.

The following table lists the names and numbers of the processes running for IIR search components.

IIR Search Component Process Name	Number of Processes
ssasrsv	2
ssalisv	2
ssacssv	2
ssacosv	2

On a Linux operating system, run the following command to verify that the total number of processes running for IIR search components is eight:

```
ps -ef | grep InformaticaIR | grep -v updsync
```

- IIR Update Synchronizer Check: Verify that the total number of processes running for Update Synchronizer is two.

On a Linux operating system, run the following command to verify that the total number of processes running for Update Synchronizer is two:

```
ps -ef | grep InformaticaIR | grep updsync
```

- IIR Test Search Client Check: Use the following steps to perform this check:

a. Run the following command to get the search client: `grep SSA_HTTPORT <PROV.TOP>/InformaticaIR/env/iss`

b. Use a browser and the following URL syntax to access the client application: `http://Your_IIR_Host_Env_IP_Address:SSA_<HTTPORT>/systems`

Here is a placeholder example: `http://xyz123.us.zzz.com:1612/systems`

- Select any system from the drop-down list and click **Next**.
 - Type an appropriate value and click **Submit Search**.
 - Note that there are no errors.
- IIR ASM Server Check: Use the following steps to perform this check:

- a. In the command line on `$InformaticaIR/bin` enter `./asmcli -b` to bring up the console.
- b. Note that the console comes up without any issues.
- c. Shut the console down.
- IIR Wiring Check: Verify that the data quality Web Services are up and running and EM is accessible. Use the following steps to perform this check:
 - a. Sign in to Oracle Enterprise Manager.
 - b. From the Navigation menu, select **Fusion Applications - Oracle Fusion Customer Relationship Management**.
 - c. Select `CrmCommonApp` and then select `CRMCommonServer_1`.
 - d. Under the Web Services section, select `DQRealTimeService` and click **Test** to launch the Web Service Tester.
 - e. On the Test Web Service Page, select **cleanseAddress** from the **Operation** drop down list.
 - f. On the Request tab, select **Custom Policy** security option and enter the Policy URL syntax `oracle/wss11_username_token_with_message_protection_client_policy` and the username and password for a user who has the **ZCH_MASTER_DATA_MANAGEMENT_APPLICATION_ADMINISTRATOR_JOB** role. For example, in development environments, this could be `MDM_ADMIN_V1`.
 - g. Under Input Arguments, select **XML View**.
 - h. Run the following blocks of code by selecting the corresponding operation from the drop-down list for each blocks of code and clicking **Test Web Service**.
 - Select the operation **cleanseAddress** from the **Operation** drop-down list and submit the following payload:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="http://xmlns.oracle.com/apps/cdm/dataQuality/realTime/publicModel/types/">
<ns1:cleanseAddress>
<ns1:input xmlns:ns2="http://xmlns.oracle.com/apps/cdm/dataQuality/realTime/publicModel/">
  <ns2:DataQualityRequestId />
  <ns2:ConfigCode>RT_LOC_CLEANSE</ns2:ConfigCode>
  <ns2:ErrorCode />
  <ns2:Message />
  <ns2:DataQualityAddress>
  <ns2:DataQualityRequestId />
  <ns2:DataQualityRecordType />
  <ns2:AddressLine1>500 oracle pky</ns2:AddressLine1>
  <ns2:AddressLine2 />
  <ns2:AddressLine3 />
  <ns2:AddressLine4 />
  <ns2:City />

```



```

<ns2:Country>US</ns2:Country>
<ns2:CountrySubEntity />
<ns2:LocationIdentifier />
<ns2:PostalCode>94065</ns2:PostalCode>
</ns2:DataQualityAddress>
</ns1:input>
</ns1:cleanseAddress>
</soap:Body>
</soap:Envelope>

```

- Select the operation **matchOrganization** from the Operation drop-down list and submit the following payload:

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="http://xmlns.oracle.com/apps/cdm/dataQuality/realTime/publicModel/types/">
<ns1:matchOrganization>
<ns1:input xmlns:ns2="http://xmlns.oracle.com/apps/cdm/dataQuality/realTime/publicModel/">
<ns2:DataQualityRequestId />
<ns2:ConfigCode>RT_ORG_DUP_PREV</ns2:ConfigCode>
<ns2:ErrorCode />
<ns2:Message />
<ns2:DataQualityOrg>
<ns2:OrganizationName>Yamaha</ns2:OrganizationName>
</ns2:DataQualityOrg>
</ns1:input>
</ns1:matchOrganization>
</soap:Body>
</soap:Envelope>

```

- Check if the responses to the above requests do not return the following `ErrorCode`:

```

<ns0:ErrorCode>ERRORED</ns0:ErrorCode>
<ns0:Message>ZCQ_CMN_ENG_CONNECT_ERR</ns0:Message>

```

If any of the responses contain this `ErrorCode`, it indicates that IIR is either not wired or is not functional.

Locking and Unlocking of Rulebase

If you are setting up IIR for matching, you must understand how the IIR Server interacts with the Oracle database. When you start an IIR server, the Rulebase Server also starts, but it does not connect to the Oracle database until the first client makes a request. This can be a search request, for example, a match call made from Oracle Fusion Applications, or, it can be an administrator request, for example, a user starting the Informatica IR Console client. A Rulebase server uses and maintains this connection unless IIR is brought down with an `idsdown` command. Once a Rulebase server connects to the Oracle database, it creates a lock that ensures that no other Rulebase server connects to the Rulebase.

When the IIR server is up and the Oracle database is shutdown or crashes, or if there is a network issue that affects connectivity, the Rulebase lock remains active. In such cases where a lock is active, all future search requests fail with an error that indicates there is another Rulebase server running.

This table lists various situations that can occur after the IIR and Oracle database connection has been established.

Possibility	Procedure
Shutting down Oracle database.	Shut down the IIR Server before shutting down the Oracle database. Else, the Rulebase is locked and it fails all future requests. Once Rulebase is locked, you need to manually unlock it.
Database crashes or there is a connectivity issue.	Unlock the Rulebase manually.
IIR server crashes or reboots due to power outage, or any similar case.	Rulebase is automatically unlocked by the first client. Requires no manual intervention.
IIR is shut down improperly, for example, with the Update Synchronizer running.	The Rulebase can become corrupt. Clean the Rulebase or create it again.

Every time an IIR server starts, a new pair of parent and child Rulebase also start. The parent server generates a new unique ID and this ID is stored in the Rulebase. In case of a Rulebase lock, if the ID generated by the parent is not equal to the ID stored in the Rulebase, the child server fails to start.

You must unlock the Rulebase manually. To do this, set an environment variable to the same ID that locks the Rulebase. Here is an example of a message that appears when a Rulebase lock is in force:

```
IS ANOTHER RULEBASE SERVER RUNNING?
```

```
Rulebase sdb:file:c:\a3i\ids\rule In use by ssa.identitysystems.com
IP=203.2.203.109 on port=1668, SSA_RB_RESTART_ID=281753650
```

For this message, enter <PROV.TOP>/InformaticaIR/bin > set SSA_RB_RESTART_ID=281753650 and then start the IIR server.

When the Rulebase server starts, it uses the environment variable to unlock the Rulebase when the two IDs match. It then uses the freshly generated parent ID to lock it again. A locked Rulebase can be unlocked only once using the same ID.

Using the Informatica IR Console Client

You can use the Informatica IR Console client for the following operations:

- For creating and indexing IIR search systems.
- Starting or monitoring the Update Synchronizer.
- Perform test searches using the search client or enabling search traces to understand or debug search behavior.
- For advanced configuration, where an existing system is edited using the System Editor to change its behavior.

These are the restrictions on using the Informatica IR Console client:

- The console client is not a multiuser application when it is used with `-a` (administrative) option, which has `read` and `write` privileges. Only single invocation of the client is possible with `-a` option. Without `-a` option, the Rulebase opens `read only`.

- Once the administration operations have been performed, there is no need for the client to be open. Close it after use. For example, after starting the Update Synchronizer, you can close the console client but this does not affect the synchronizer.
- If the Informatica IR Console client is left connected for several days and there is a database or network interruption, the following situations can occur:
 - If the Update Synchronizer fails, it cannot be restarted through the client unless the client is restarted.
 - Client loses all functionality except for **Tools** and **Search Client** unless restarted.
 - You do not need to restart the IIR server for restarting client connection.

Encrypting the Informatica Identity Resolution Dictionary-alias File: Explained

Clients such as Console Client or Fusion Real-Time and Batch that connect to IIR, pass a mandatory parameter called rulebase name. It is an alias for the connection string required by IIR to connect to the FUSION_DQ schema in the Fusion Application Database. The alias is defined in a file called `ssadb.dic` in `<PROV.TOP>/InformaticaIR/ids` directory.

The following example shows how a typical `ssadb.dic` would look like.

```
rtunitrb odb:0:fusion_dq/fusion_dq@rws65311
fusion_s01 odb:1:fusion_dq/fusion_dq@rws65311
btunitrb odb:1:fusion_dq/fusion_dq@rws65311
fusion_s02 odb:2:fusion_dq/fusion_dq@rws65311
zcqsrc odb:99:fusion_dq/fusion_dq@rws65311
```

The alias `rtunitrb` is the alias for the rulebase name. The remaining aliases are used for different purposes. For example, `fusion_s01` is used to create the FusionDQRealtime System, `zcqsrc` is used by the loader process to get source data from Oracle Fusion and so on.

Note

The setup mentioned here is not mandatory and should be used only based on use cases, and in a specific deployment scenario.

Changing Rulebase Names

In Oracle Fusion, the predefined alias for the rulebase name is `rtunitrb`. If there is ever a need to change the rulebase name, you can change it in the dictionary file, and ensure that the change is also carried over to the Server Operation Page (used in IIR Matching). For example, if `rtunitrb` is changed to `foo`, then the rulebase alias in the Server Operation Page should be changed from `ssa:rtunitrb` to `ssa:foo`. If you made this change while the server was up and running, you need to restart the IIR Server to bring the change into effect.

Encryption ssadb.dic

It is recommended to create a backup copy of the ssadb.dic file and store it as a template for future use, before encrypting it. It is also recommended to replace the password on the backup copy with an empty string. You can later provide a password for this backed up copy, if you intend to use it.

Access the directory where PERL is available, either using \$PATH or the absolute path and do the following:

- On Windows, run the following commands:
 - PROV.TOP\InformaticaIR\bin > isss.bat
 - PROV.TOP\InformaticaIR\bin > perl convfusiondict.pl <password>
- On Oracle Enterprise Linux, run the following commands:
 - PROV.TOP/InformaticaIR/bin >. .setfusionEnv.sh

Note

If using C-SHELL, use `source setfusionEnv.csh` instead of `setfusionEnv.sh`

-
- PROV.TOP/InformaticaIR/bin > perl convfusiondict.pl <password>

Note

In these commands, <password> can be any string that the administrator chooses. Password protects the dictionary file from being edited (either to add a new entry or delete an existing entry). In Oracle Fusion, this password is a place holder to run the iirdict utility.

Changing Password after Encryption

After ssadb.dic is encrypted, there may be occasions where the password for FUSION_DQ might be changed by a database administrator, as part of security due diligence.

To change the password for ssadb.dic, do the following.

1. Take a backup of the existing ssadb.dic (as mentioned in the earlier section).
2. Delete the existing ssadb.dic (in use).
3. Enter the new password for the backup up file and rename the file to ssadb.dic.
4. Encrypt the file as mentioned earlier.

Manually Installing Informatica Identity Resolution: Procedures

This help topic describes the provisioning and platform support, software requirements, and the steps for the manual installation of Informatica Identity Resolution (IIR).

Assessing Provisioning and Platform Support

Use this information to determine whether you need to install IIR manually.

When CRM is provisioned using the provisioning framework, an instance of IIR (64 Bit) is provisioned on the following platforms: OEL 64 Bit, Windows 64 Bit, AIX 64 Bit, Solaris Sparc 64 Bit.

However, IIR is not provisioned as part of CRM Provisioning for the Solaris X-64 Bit platform. For this platform, install IIR manually on any of the supported platforms such as OEL 64 Bit. Except for the manual install, IIR setup steps for Solaris X-64 Bit platform are identical to those for other platforms such as OEL 64 Bit.

Assessing Software Requirements

Use this information to ensure compliance with the software requirement for the manual installation of IIR.

The host on which IIR is installed should have access to an Oracle DB Client for connecting to the Fusion Application Database.

The host should have its `GNU libc` version 2.5 or later.

The size of `FUSION_TS_DQ` (the default table space for the Oracle Fusion data quality and hence the repository for IIR), should have a table space size at least 10 times the size of the data. The data here represents the fields used in `FusionDQ Realtime.sdf`.

Installing Informatica Identity Resolution

Use the following steps to manually install IIR on the OEL 64 Bit platform.

1. Unzip the following files in an appropriate location
 - `IIR_901sp1_linux_amd64.zip`
 - `Fusion-Exts-IIR-901sp1.zip`

Note

This file resolves to `fusion_iir` after unzipping.

2. Go to the `fusion_iir` directory and edit the `install.props` file to make sure the properties described in the following table are populated correctly.

Note

This table describes the properties and their values that must be populated correctly for installing IIR manually. None of the properties except the ones mentioned in this table should be modified.

Properties	Value Descriptions (To be Replaced with Actual Values)	Sample Values
ORACLE_HOME	Oracle home for DB client used by IIR	
TNS_ADMIN	TNS_ADMIN for DB client used by IIR	
LD_LIBRARY_PATH	Library path on IIR host	
FUSION_STAGE_DIR	Directory where Fusion Ext Zip is unzipped	
IIR_STAGE_DIR	Directory where IIR Zip is unzipped	
IIR_VERSION	Name of the version to which that the IIR unzip resolves	IIR_901sp1_linux_amd64
IIR_DB_HOSTNAME	Fusion application DB host	
IIR_DB_PORT	Fusion application DB port	
IIR_DB_SID	Fusion Application DB SID	
IIR_DB_USER	fusion_dq	
IIR_DB_PASSWD	Credentials for fusion_dq	
IIR_INSTALL_LOG_LOC	Location to which Install logs should go	
IIR_INSTANCE_1_PORT	Start port for the license server	1660 (if available, this port should work fine as the default port)
INSTALL_IIR_OBJECTS	Yes or No	No (if IIR is installed as a Secondary Server; for more information, see: Setting up Multiple Informatica Identity Resolution Instances: Overview)

- Execute the following command after changing to the `fusion_iir` directory: `runInstaller.sh install.props.`

It typically takes about 8 to 12 minutes to install IIR.

Note

Use the same steps to install IIR on the Windows 64 Bit platform, except for running `runInstaller.bat` in `fusion_iir`

Manage Server Configurations

Server Configurations: Explained

Server configurations are predefined configurations for third-party data quality servers.

You can search, review, and edit server configurations. Although you can edit the configuration name, server address, server port, and values of some of the configuration parameters, you cannot add or delete a server configuration parameter.

There are four predefined server configurations:

- Real-Time and Batch Basic Match Server
- Real-Time Cleanse Server
- Batch Cleanse Server
- Advanced Batch Match Server

Real-Time and Batch Basic Match Server

Used for basic real-time duplicate prevention and batch data matching and duplicate identification. This configuration requires you to synchronize the Oracle Fusion Trading Community Hub registry data with the third party data quality engine repository periodically to ensure appropriate matching indexes are updated continuously.

Real-Time Cleanse Server

Used for real-time data cleansing operations that happen at touch point applications, such as Oracle Fusion Sales and Oracle Fusion Customer Center, and so do not need synchronization.

Batch Cleanse Server

Used for batch cleansing operations during which records in a batch are sent to the data quality server for cleansing one at a time, and the data quality engine returns the cleansed record to the registry likewise, one at a time, thus completing the loop.

Advanced Batch Match Server

Used for advanced batch matching that is done to find duplicates amongst a group of records before uploading them to the Oracle Fusion Trading

Community Hub registry. For example, an acquiring company may do an advanced batch match of the legacy data of the acquired company before uploading the data to the registry. In such a case, data is uploaded to the repository of the data quality engine directly for duplication identification and after the duplication identification processing it is imported to the Oracle Fusion Trading Community Hub registry.

Oracle Fusion Data Quality Connector Setup: Procedures

Once you complete the Informatica Identity Resolution (IIR) matching and cleansing setups, you must ensure that Oracle Fusion Applications is connected to the data quality engine.

Connecting Oracle Fusion Applications to Data Quality Engine

To connect Oracle Fusion Applications to the data quality engine:

1. Map Oracle Fusion Applications to IIR Matching server.
 - a. Log in to the Functional Setup Manager application as an administrator user.
 - b. Under **Assigned Implementation Tasks**, search for `Manage Server Configuration` setup tasks.
 - c. Select **Go to Task**. This opens the task-flow page.
 - d. Click **Search** to find all existing configurations.
 - e. In the **Search Results**, select **Realtime and BatchBasic Server**.
 - f. Select **Actions - Edit**.
 - g. Enter the **Server Address** and **Server Port** of the IIR Matching Server. The default port is 1666. To know the current port, use `grep SSA_SEPORT=${PROV.TOP}/InformaticaIR/env/iss`.
 - h. Select **Enable Matching**.
 - i. (Optional) Select **Enable High Availability**.
 - j. Click **Save and Close**.
2. Map Oracle Fusion Applications to IIR Cleansing server.
 - a. Log in to the Functional Setup Manager Application as an administrator user.
 - b. Under **Assigned Implementation Tasks**, search for `Manage Server Configuration` setup tasks.
 - c. Select **Go to Task**. This opens the task-flow page.
 - d. Click **Search** to find all existing configurations.
 - e. In the **Search Results**, select **Realtime Cleanse Server**.
 - f. Select **Actions - Edit**.

Note

Repeat the steps to select Batch Cleanse Server from the search results and edit it.

-
- g. Enter the **Server Address** and **Server Port** of the IIR Cleansing server. The default port is 1666. To know the current port, use `grep SSA_SEPORT=${<PROV.TOP>/InformaticaIR/env/iss}`.
 - h. Select **Enable Cleansing**.
 - i. (Optional) Select **Enable High Availability**.
 - j. Click **Save and Close**.
-

Note

By default, the IIR server instance is the same for both matching and cleansing, but you can configure separate instances if required.

Manage Data Quality Engine Matching Setup

Informatica Identity Resolution Matching Setup: Procedures

When you use the configurations provided by Oracle Fusion Trading Community Data Quality, matching consists of initial indexing and index synchronization.

Initial Indexing

Perform the following for initial indexing:

1. Log in to the Informatica Identity Resolution (IIR) Host machine.
2. Enter `cd <PROV.TOP>/InformaticaIR/bin`.
3. Run `setfusionEnv.sh`.
4. Start the Informatica IR Console using `./idsconc -a` option.
5. On the launched console, select **Rulebase Type** as `SSA` and **Alias** as `rtunitrb`.
6. Click **OK**.
7. Select **Yes** for a new Rulebase.
8. In Informatica IR Console, navigate to **System - New - Create a system from SDF**.

Enter the following information:

- **System Name:** `FusionDQRealtime`.
- **System Definition File:** `<FULLPATH>/InformaticaIR/ids/FusionDQRealtime.sdf`.
- **DB Type:** `SSA`.
- **Alias:** `fusion_s01`.

9. Go to **System** and select **System Name** as FusionDQRealtime.
10. Click **OK**.
11. Go to **System - Load IDT**.

Use **Loader Definition** list to load the following ID tables one by one:

- load_location.
 - load_organization.
 - load_org_address.
 - load_person.
 - load_per_address.
 - load_per_phone.
12. Click **OK** for each ID table.

The IIR load jobs launched from the Informatica IR Console pull the data from the Oracle Fusion Applications database and create keys for them. It then prepares the data in the IIR Server and uses SQL loader to load the data back into the Oracle Fusion Applications database. During this process, if any load job fails and a fix has been applied, you must delete the corresponding .dat file from the PROV.TOP//InformaticaIR/ids directory and then refresh the applicable ID table.

To refresh an ID table:

1. Go to **System - Refresh**.
2. Select **Refresh Selected Objects**.
3. Choose the ID table name to refresh.
4. Click **OK**.

Index Synchronization

Once initial indexing is done, you can continue with adding or editing records, but you must synchronize the index as follows:

1. Start the Update Synchronizer on IIR.

Note

Refer to Starting Up and Shutting Down Update Synchronizer section of this help topic.

-
2. Start the Oracle Fusion Synchronization Oracle Enterprise Scheduler job.

Note

Refer to Running the Schedule Synchronization Process section of the following help topic Synchronizing Trading Community Registry and Data Quality Engine Repository Data: Worked Example.

Starting Up and Shutting Down Update Synchronizer

The Update Synchronizer is a batch job that runs in the IIR Server. Primary function of an Update Synchronizer is to take records in the no source access (NSA) table periodically, create the keys for them, and synchronize these to identity tables (IDTs) and identity indexes (IDXs). The records in NSA are created by the Oracle Fusion Synchronization Oracle Enterprise Scheduler job. Once the Update Synchronizer has completed the job, it would clear the records from NSA.

Starting Up Update Synchronizer: You can start the Update Synchronizer from the Informatica IR Console.

1. Log in to Informatica IR Console.
2. Go to **System** and select **System Name** as `FusionDQRealtime`.
3. Go to **Tools - Synchronizer - Run Synchronizer**.
4. Click **OK**.

Select the **All** option for **IDT Name** and retain the rest of the defaults.

5. Click **OK**.

Shutting Down Update Synchronizer: Once started, the Update Synchronizer can be left running for the lifecycle of the IIR Server unless there is a need to shut it down.

1. Log in to Informatica IR Console.
2. Double click **Update Synchronizer** under **Launched Jobs** to open the Synchronize window.
3. Select **Yes**. The Synchronizer terminates within 60 seconds.
4. Close the Synchronize window after shutdown.

When there is loss of connectivity with the database, Update Synchronizer stops functioning. You must then restart it.

Extending Predefined IIR Matching Configurations: Procedures

This topic describes how to extend predefined Informatica Identity Resolution (IIR) matching configurations to overcome the skewness of matching results when fields or attributes of the records being matched are null.

Note

The ability to extend predefined IIR matching configurations is currently not available in Oracle Cloud implementations.

When a record is sent for matching to Informatica Identity Resolution (IIR), the engine returns the final set of matches, by performing a two-step process. The consecutive steps are listed below:

1. Search or Candidate Selection: As part of this step, IIR searches and shortlists candidate records from among the repository records (also called file records).
2. Matching or Scoring: During this step, IIR does a field by field level comparison of the candidate record against the corresponding field in the record being matched (also called the searched record). The individual score of a field is: $\text{weight} * \text{score}$. The total score is the sum of all individual scores.

Note

For more information on the IIR matching process and predefined matching configurations, refer the IIR vendor documentation in <PROV.TOP>/InformaticaIR/pdf directory within the IIR Installation.

Since the total score is the sum of all individual scores, scoring produces skewed results if an attribute or field in either the searched record (the record being matched) or in the candidate record (the file record) is null, because the weight contribution of such a field would be 0.

The following table provides an example of how scoring results can be skewed, if an attribute or field in either the searched record (the record being matched) or in the candidate record (the file record) is null.

Searched Record (the record being matched)	File or Candidate Record	Score
Lawrence, 105 E. Graham Ave	Lawrence	$(100 * 800) / 800 = 100$
Lawrence	Lawrence, 105 E. Graham Ave	$(100 * 800) / 800 = 100$

In this example the score is 100 percent even when the address field is missing, because predefined IIR matching configurations ignore weight contributions for any null attribute or field in either the searched record (the record being matched) or in the candidate record (the file record).

To overcome the skewness of matching results when fields or attributes of the records being matched are null, Oracle Fusion Data Quality ships a set of custom rulebase population, which when used, ensures that the weight for any attribute is 0, only if that attribute is null for the searched record (the record being matched).

The following table provides an example of scoring done using the custom rulebase population that ensure that the weight for any attribute is 0, only if attribute is null for the searched record (the record being matched).

Searched Record (the record being matched)	File or Candidate Record	Score (predefined population)	Score (custom population)
Lawrence, 105 E. Graham Ave	Lawrence	$(100 * 800) / 800 = 100$	$\{(100 * 800) (0 * 200)\} / \{800 + 200\} = 80$
Lawrence	Lawrence, 105 E. Graham Ave	$(100 * 800) / 800 = 100$	$(100 * 800) / 800 = 100$

Procedure for Extending Predefined IIR Matching Configuration

Use this procedure to extend predefined Informatica Identity Resolution (IIR) matching configurations.

1. Log in to **IIR Admin Console** and select `FusionDQRealtime` system.
2. Stop the **Update Synchronizer** if it is running.
3. Start **System Editor**.
4. Select **Edit** and select **Yes, this time only** or **Yes, do not ask me again for this session** as appropriate.
5. Select **Org-dup** (under Search) in the System Editor.
6. Select the **Score Logic** tab, and on the **Controls** field add `Fusion_right` after the `Purpose=` In other words, after the change, the `Match Purpose` should be `Fusion_Organization`, as opposed to `Organization`.
7. Click **Load**. Close the **Edit System** pop up once you see the message `Job completed`, and repeat Step 5 for all the remaining 8 searches from `location_dup` through `person_dup` in the **System Editor**.
8. (Recommended) Restart the IIR to avoid caching and to ensure that the new scoring mechanism is used.

Note

- The out of the box predefined configurations use the US population: `usa.ycp`. To create other custom populations, select the relevant population out of the 64 populations from the zipped file `OracleCustomPopulations.zip` in the `InformaticaIR/pr/default` directory and customize it.
 - To revert to the out of the box predefined population, undo the steps above, by removing `Fusion_` from the `Match Purpose`, from all the nine searches in the **System Editor** and by doing a load using the Step 7.
-

Manage Matching Configurations

Matching Configurations: Explained

Matching configurations are predefined configurations used for real-time and batch matching of party entities with the intention of preventing and identifying duplicate entries in the trading community registry.

Real-time and batch matching are available for the following trading community entities:

- Organization
- Person
- Location

There are six predefined matching configurations, three each for real-time and batch matching.

These configurations include both Oracle Fusion Trading Community Data Quality parameters and the parameters of the embedded data quality engine, Informatica Identity Resolution (IIR). While some of the parameters, such as **MatchTolerance**, are single valued, others, such as **Manage Search Definition**, are multi-valued.

You can search, review, and edit these seeded matching configurations.

Note

Although you cannot add or delete a matching configuration parameter, you can change the configuration name and update the parameter values to suite your requirements. For more information, see: Matching Configuration Parameters.

Real-time Matching Configurations

The following are the predefined real-time matching configurations:

- Real Time Location Duplicate Prevention
- Real Time Organization Duplicate Prevention
- Real Time Person Duplicate Prevention

Real-time matching prevents individual entry of duplicate trading community entities, such as organization, person, or location, through calling touch point applications, such as Oracle Fusion Sales and Oracle Fusion Customer Center into the trading community registry. Real-time matching finds all possible duplicate records that may exist in the registry for an entered record, and assigns a match score to each potential duplicate identified. Based on the match score returned by the service and the threshold settings in the configuration, the calling application can provide the option to either select an existing duplicate record or continue to create a new record.

Batch Matching Configurations

The following are the predefined batch matching configurations:

- Batch Location Basic Duplicate Identification
- Batch Organization Basic Duplicate Identification
- Batch Person Basic Duplicate Identification

Batch matching leads to the identification of duplicate entries in the trading community registry, between trading community registry and sets of data, such as import batches, or within sets of data to resolve them by merging or linking. While real-time matching provides duplicate prevention for only one record and returns potential matches for the record being entered, the batch matching service takes a set of records of the same type, such as person, organizations, or locations and identifies all possible matches within these sets of records. The duplicate data is presented in sets of possible matches with a score associated with each individual record in the set.

IIR Matching Process

Based on the matching configurations provided by Oracle Fusion Trading Community Data Quality, the embedded data quality engine, Informatica Identity Resolution (IIR), returns the final set of matches by performing a two-step process comprising candidate selection and scoring.

These steps are described below:

1. **Candidate Selection:** This step requires that the matched-against records are stored in the IIR repository and indexed. IIR uses search definitions to search and shortlist candidate records from among the repository records.
2. **Scoring:** In this step, IIR does a field by field level comparison of the candidate record against the corresponding fields in the record being matched. The matching fields and the scoring are also defined in the search definition.

Defining Indexes and Index Synchronization

Matching using the configurations provided by Oracle Fusion Trading Community Data Quality requires both defining indexes, or initial indexing, and index synchronization of the Informatica Identity Resolution (IIR) repository with the trading community registry.

A prerequisite to defining indexes or initial indexing is creation of a new system on IIR using the system definition file, `FusionDQRealtime.sdf`, shipped with the application that uses US population and contains information about IIR match rules and indexes.

Before defining indexes, review the index information in the out of the box system definition file `FusionDQRealtime.sdf` to verify that it is up to date and make any modifications required to suite your specific business needs.

Staging Tables

IIR staging tables, also called identity tables (IDTs), are used to achieve the following:

- Stage Oracle Fusion Applications data.
- Partition the data based on trading community entities and their attributes.

Identity tables (IDTs) are also used for periodic index synchronization of the third party data quality engine repository with the Oracle Fusion Trading Community Hub registry to account for the continual updates made to the registry.

Note

IIR staging tables definitions are stored in `FusionDQRealtime.sdf` shipped with the application.

The following table describes the different staging or identity (IDT) tables.

Entity	IDT Name	Data Content	Comments
Organization	ORG_PRIMARY_IDT	Profile information and primary attributes, such as primary address and primary phone, of an organization.	The IDT includes one row for each organization.
Organization	ORG_ADDRESS_IDT	Multiple addresses and primary attributes, such as primary phone, of an organization.	IDT does not include organizations that have no addresses. For organizations that have addresses, the IDT includes one row for each active address.
Person	PER_PRIMARY_IDT	Profile information and primary attributes, such as primary address and primary phone, of a person.	The IDT includes one row for each person.
Person	PER_ADDRESS_IDT	Multiple addresses and primary attributes, such as primary phone, of a person.	IDT does not include persons that have no addresses. For persons that have addresses, the IDT includes one row for each active address.
Person	PER_PHONE_IDT	Multiple Phones and Primary attributes (primary phone) of a Person, with Key created on "Concatenated Address Lines".	IDT does not include persons that have no phones. For Persons that have phones, the IDT includes one row for each active phone.
Location	LOCATION_IDT	Location Information	The IDT includes one row for each active location.

Search Definitions

IIR uses search definitions to search and shortlist candidate records from among the repository records.

The following table describes the different search definitions

Entity	Search Definition	Functionality
Organization	org-dup	Uses Organization Name as key or acquisition attribute and scores on other fields, using weights. Organization Name has the highest weight (about 67%), although other attributes contribute, as well.

Organization	org-address-dup (multi-search definition)	Leads to two searches, one against Organization Name (key 1) and another against Address Line 1 (key 2). So matching on any of these yields a match.
Person	person-dup	Uses Person Name as key or acquisition attribute and scores on other fields, using weights. Organization Name has the highest weight (about 67%), although other attributes contribute, as well.
Person	person-address-dup (multi-search definition)	Leads to two searches, one against Person Name (key 1) and another against Address Line 1 (key 2) and combines results. So matching on any of these yields a match.
Person	person-phone-dup (multi-search definition)	Leads to two searches, one against Person Name (key 1) and another against Phone (key 2) and combines results. So matching on any of these yields a match.
Location	location-dup	Uses Address Line 1 as the key or acquisition attribute and scores on other fields, using weights. Address Line 1 has the highest weight (about 67%).

Match Options Behavior

When finding out duplicate organization and person parties, some users might want to match potential duplicate records using multiple addresses and phones. However, indexing multiple records with multiple addresses and phones can adversely affect performance. Oracle Fusion Trading Community Data Quality tries to strike a balance between the accuracy and speed of duplicate identification by taking the following optimal approach:

1. Partitioning the Oracle Fusion Applications data based on trading community entities and their attributes in staging tables before loading the data into the IIR repository.
2. Providing appropriate match options on the configuration page.

The following table describes real time matching behavior for the different available match options.

Entity	User Options	API or Web service Behavior
Organization	Match Against: <ul style="list-style-type: none"> • Primary Only • Primary OR All Addresses 	For Primary Only match option, service goes against <code>PRIMARY search def</code> For Primary OR All Addresses match option, service goes against <code>ADDRESS search def</code> , if address of the record being matched is not null, else goes against the <code>PRIMARY search def</code>

Person	<p>Match Against:</p> <ul style="list-style-type: none"> • Primary Only • Primary OR All Addresses • Primary OR All Phones • Primary OR All Addresses OR All Phones (with a user specified tiebreaker, when the record being matched has both address and phone) 	<p>For Primary Only match option, service goes against PRIMARY search def.</p> <p>For Primary OR All Addresses match option, service goes against ADDRESS search def, if address of input record is not null, else goes against the PRIMARY search def.</p> <p>For Primary OR All Addresses match option, service goes against PHONE search def, if phone of the record being matched is not-null, else goes against the PRIMARY search def.</p> <p>For Primary OR All Addresses OR All Phones match option:</p> <ul style="list-style-type: none"> • If both Address and Phone of the record being matched are not null, the service goes against ADDRESS OR PHONE search def based on tiebreaker logic. • If Address is not null, the service goes against ADDRESS search def. • If Phone is not null, the service goes against ADDRESS search def • If both Address and Phone are null, the service would go against the PRIMARY.
--------	--	--

The following table describes batch matching behavior for the different available match options.

Entity	User Options	API or Web service Behavior
Organization	<p>Match Against:</p> <ul style="list-style-type: none"> • Primary Only • Primary AND All Addresses 	<p>For Primary Only match option, service goes against only ORG PRIM</p> <p>For Primary AND All Addresses match option, service goes against both ORG PRIM and ORG ADDRESS.</p>

Person	Match Against: <ul style="list-style-type: none"> • Primary Only • Primary AND All Addresses • Primary AND All Phones • Primary AND All Addresses AND All Phones 	For Primary Only match option, service goes only against PER PRIM For Primary AND All Addresses match option, service goes against both PER PRIM and PER ADDRESS. For Primary AND All Phones match option, service goes against both PER PRIM and PER PHONE. For Primary AND All Addresses AND All Phones match option, service goes against all: PER PRIM and PER ADDRESS and PER PHONE.
--------	--	--

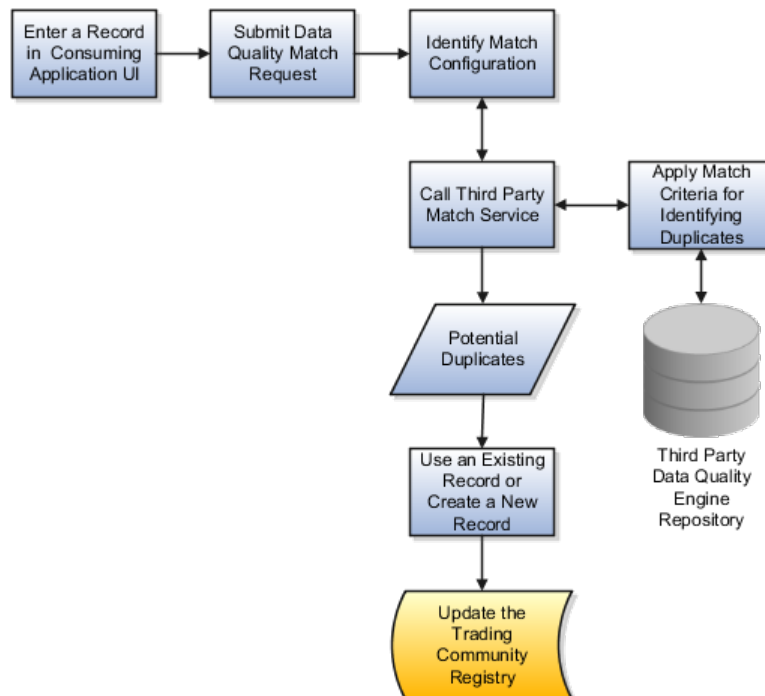
Real-Time Duplicate Prevention Components: How They Work Together

Real-time duplicate prevention enables you to prevent the individual entry of duplicate trading community entities, such as organization, person, or location, in the trading community registry through Oracle Fusion Trading Community Data Quality matching services consuming applications, such as Oracle Fusion Customer Center.

Real-time duplicate prevention comprises the following components:

- Real-Time Matching Service
- Data quality service consuming applications
- Matching configurations
- Server configurations

This figure shows the real-time duplicate prevention components in action.



Real-Time Matching Service

During the data entry process, the Real-Time Matching Service finds all possible duplicate records that may exist in the trading community registry for an entered record, and assigns a match score to each potential duplicate identified.

Consuming Applications

Real-time duplicate prevention finds use in data quality service consuming applications such as Oracle Fusion Receivables and Oracle Fusion Customer Center. When users try to enter a new person, organization, or location record through their UI into the Oracle Fusion trading community registry, the service finds all possible duplicate records that may exist in the registry for an entered record, and assigns a match score to each potential duplicate identified. Based on the match score returned by the service and the threshold settings in the configuration, the calling application can provide the option to either select an existing duplicate record or continue to create a new record.

Matching Configurations

Matching configurations are predefined configurations used for real-time and batch matching of party entities with a view to preventing and identifying duplicate entries in the trading community registry. Oracle Fusion Trading Community Data Quality has three predefined real-time matching configurations:

- Real Time Location Duplicate Prevention
- Real Time Organization Duplicate Prevention
- Real Time Person Duplicate Prevention

You can search, review, and edit these seeded matching configurations. While you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements.

Server Configurations

Real-time duplicate prevention makes use of the Real-Time and Batch Basic Match Server.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing. Server configurations show all the parameters that can be set at either the matching configuration level or the server configuration level and provide the ability to modify the values of the server configuration level parameters. The parameters that are set at the server level are applicable to all the matching configurations.

Batch Duplicate Identification Components: How They Work Together

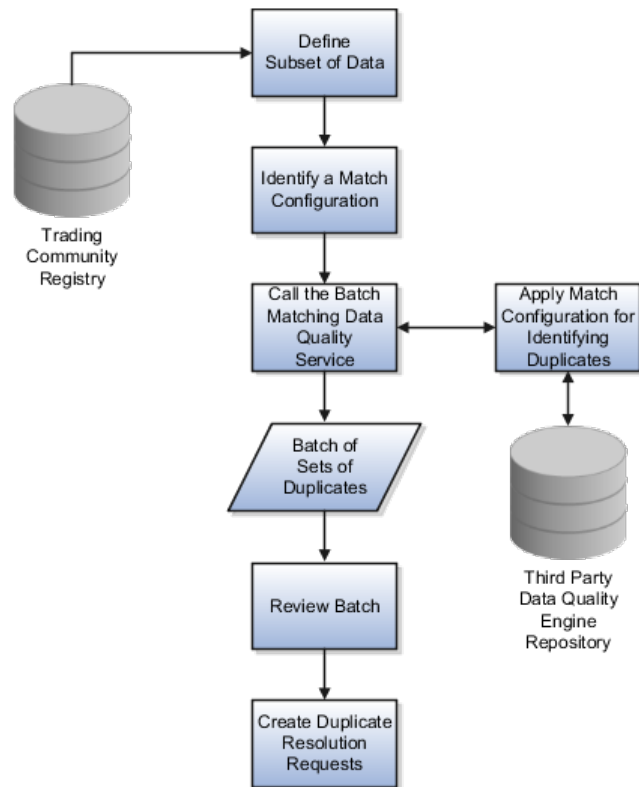
Batch duplicate identification enables you to identify duplicate entries in the trading community registry, between trading community registry and sets of data, such as import batches, or within sets of data to resolve them by merging or linking.

Batch duplicate identification comprises the following components:

- Batch Matching Service
- Data quality service consuming applications

- Matching configurations
- Server configurations

This figure shows the batch duplicate identification components in action.



Batch Matching Service

The Batch Matching Service is used for identification of potential duplicates for trading community entities such as persons, organizations, and locations.

Consuming Applications

The batch duplicate identification capability finds use in master data management applications such as Oracle Fusion Trading Community Hub, and other data quality service consuming applications that import data in batches. The capability takes a set of records of the same type, such as persons, organizations, or locations and identifies all possible matches within these sets of records. The duplicate data is presented in sets of possible matches with a score associated with each individual record in the set. These sets of potential duplicates can then be resolved by merging or linking.

Matching Configurations

Matching configurations are predefined configurations used for batch and real-time matching of party entities with a view to identifying and preventing duplicate entries in the trading community registry. Oracle Fusion Trading Community Data Quality has three predefined batch matching configurations:

- Batch Location Basic Duplicate Identification
- Batch Organization Basic Duplicate Identification
- Batch Person Basic Duplicate Identification

You can search, review, and edit these seeded matching configurations. While you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements.

Server Configurations

Batch duplicate prevention makes use of the Real-Time and Batch Basic Match Server.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing. Server configurations show all the parameters that can be set at either the matching configuration level or the server configuration level and provide the ability to modify the values of the server configuration level parameters. The parameters that are set at the server level are applicable to all the matching configurations.

Matching Configuration Parameters

Matching configuration parameters are system-level parameters that control aspects of the Oracle Fusion Trading Community Data Quality matching services. The Oracle Fusion Trading Community Data Quality matching services read values from the ZCQ_CONFIG_PARAM_VALUES and ZCQ_SO_PARAM_VALUES tables.

Matching configurations are predefined configurations that include parameters from both Oracle Fusion applications and the integrated third-party data quality engine. Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements. The text describes only the generic Oracle Fusion Trading Community Data Quality parameters and their values. For third-party data quality engine specific parameters and their values, see relevant third-party documentation.

Real Time Location, Organization, and Person Duplicate Prevention Parameters

The following Oracle Fusion Trading Community Data Quality parameters control matching operations for preventing individual entry of duplicate trading community entities, such as location, organization, and person through calling touch point applications, such as Oracle Fusion Sales and Oracle Fusion Customer Center into the trading community registry.

MaxNumReturn

Parameter Description: Controls the maximum number of matched records returned by the data quality real-time matching service.

ScoreThreshold

Parameter Description: Determines the score above which the matched records are returned by the matching service. Records equal to or greater than the score are considered as matches and the records with scores less than the threshold are rejected.

Batch Location, Organization, and Person Basic Duplicate Identification Parameters

The following Oracle Fusion Trading Community Data Quality parameters control matching operations for identification of duplicate entries in the trading

community registry, between trading community registry and sets of data, such as import batches, or within sets of data to resolve them by merging or linking.

MatchMode

Parameter Description: Determines how the records in the batch are matched during batch matching process. Select Exhaustive to include in the batch match process even the records already showing up in previously matched groups. Select Optimized to skip such records.

NumChildJobs

Parameter Description: Determines the number of child requests spawned for a batch matching request. The child requests are useful when there is a need to process data in parallel.

ScoreThreshold

Parameter Description: As described earlier.

Manage Data Quality Engine Cleansing Setup

Informatica Identity Resolution Address Cleansing Setup: Procedures

Oracle Fusion Applications Installer installs a demo version of AddressDoctor postal reference data and product license keys that you can use to test address cleansing functionality; however, this demo is restrictive and provides you with a small subset of the complete functionality. For using the complete functionality, you must request for production license and postal reference data.

Using Demo License and Demo Postal Reference Data

If you are using demo keys and data, no specific setup is required for address cleansing server. You can quickly verify the setup using `PROV.TOP/InformaticaIR/bin > asmcli -b`. When successful, it launches the Address Standardization Module (ASM) client.

Using Production License and Postal Reference Data

AddressDoctor postal reference data and product license keys are not sold or distributed by Oracle. To acquire a license key and postal reference data, contact Informatica/AddressDoctor by sending an e-mail to OracleAV@informatica.com. The AddressDoctor license key restricts the use of address validation to the purchased countries or territories and represents a twelve (12) month subscription to the postal reference data. Once AddressDoctor receives the e-mail request through OracleAV@informatica.com, an AddressDoctor representative will contact you to define the requirements and complete the transaction. Once finalized, AddressDoctor will provide the license key and download credentials/instructions for accessing the postal reference data. Notification of regular updates to the postal reference data will be provided through e-mail.

Once the production license key and postal reference data are obtained, perform the following:

- `PROV.TOP/InformaticaIR/bin > idsdown.`
- `PROV.TOP/InformaticaIR/bin > cd PROV.TOP/InformaticaIR/ssaas/ad5/ad/db.`

- `PROV.TOP/InformaticaIR/ssaas/ad5/ad/db > cp mylocation/key.` Here, `key` is a text file that contains the license provided by AddressDoctor.
- `PROV.TOP/InformaticaIR/ssaas/ad5/ad/db > cp mylocation/*.MD].` Here, `/*.MD]` stands for the postal reference data file. AddressDoctor supports 248 countries and each MD file is per country or a group of countries. Each of these files should be copied to this directory.

Manage Cleansing Configurations

Cleansing Configurations: Explained

Cleansing configurations are predefined configurations used for real-time and batch cleansing of the location entity.

There are two predefined cleansing configurations, one each for real-time and batch matching:

- Real Time Data Quality Location Cleanse
- Batch Data Quality Location Cleanse

You can search, review, and edit these seeded matching configurations.

Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements.

These configurations include parameters from both Oracle Fusion applications and the integrated address cleansing engine. While some of the parameters are single valued others are multi-valued.

Real-Time Cleansing

Real-time cleansing endows Oracle Fusion applications with an online, interactive service to cleanse, standardize, and validate addresses during the data entry process either through a UI or any other service creating address data into the Oracle Fusion Trading Community Hub registry.

Batch Cleansing

Batch cleansing enables you to perform address cleansing, standardization, and validation for a subset or entirety of the location records in the registry, or as part of a data import process. It also enables you to ensure data accuracy of the Oracle Fusion Trading Community Hub registry. The need to cleanse the registry data arises owing to data decay over time and the need to resolve inherited issues from consolidated data in the registry. For example, postal codes and city boundaries change over time and require regular address cleansing to ensure that the addresses in the registry are correct, validated, and standardized at all times. Besides, historic incoming address data may have originated from multiple sources with each source system following different data storage formats and norms and the addresses might not have been cleansed during the import or consolidation process.

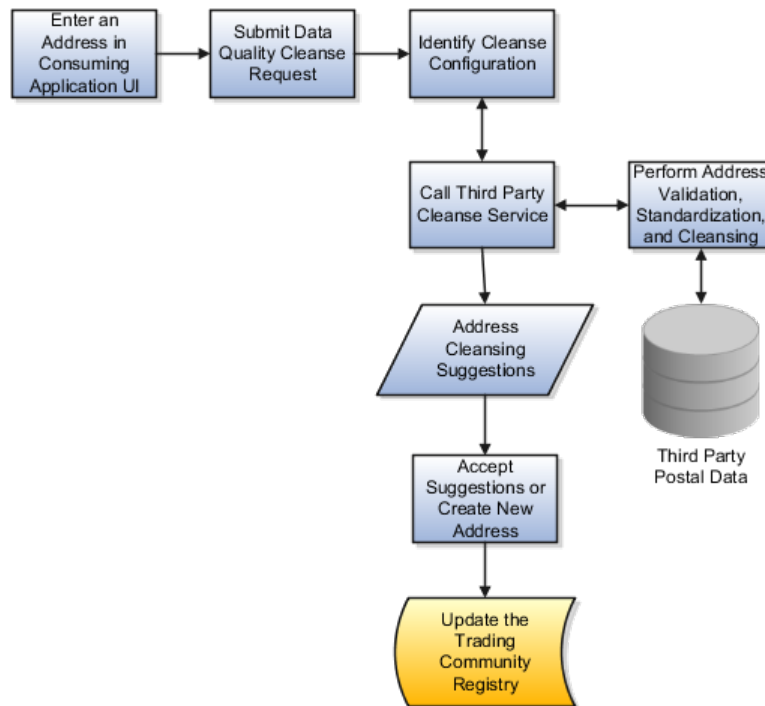
Real-Time Address Cleansing Components: How They Work Together

Real-time cleansing provides consuming applications an online, interactive service to cleanse, standardize, and validate addresses during the data entry process.

Real-time address cleansing comprises the following components:

- Real-Time Cleansing Service
- Data quality service consuming applications
- Cleansing configurations
- Server configurations

This figure shows the real-time address cleansing components in action.



Real-Time Cleansing Service

The Real-Time Cleansing Service performs validation, standardization, and cleansing of addresses during data entry into the trading community registry.

Consuming Applications

Real-time address cleansing finds use in data quality service consuming applications such as Oracle Sales and Oracle Fusion Customer Center. When a user tries to enter a new address record through the UI of these applications or through any other service creating address data into the Oracle Fusion trading community registry, the service validates the entered address data against country-specific postal service lists, such as United States Postal Service. If there are any issues with the entered address data, the service returns a list of possibilities from which the consuming application user can select an appropriate one or continue to create a new address record.

Cleansing Configurations

Cleansing configurations are predefined configurations used for real-time and batch cleansing of the location entity.

Oracle Fusion Trading Community Data Quality has a predefined real-time cleansing configuration named Real Time Data Quality Location Cleanse. The

real-time address cleansing service uses this cleansing configuration to derive the default country to be used to perform validation when address parsing fails to find a valid country within the address.

You can review and edit this predefined cleansing configuration. While you cannot add or delete a configuration parameter, you can change the configuration name and update the parameter values to suite your requirements.

Server Configurations

Real-time cleansing makes use of the Real-Time Cleanse Server configuration.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

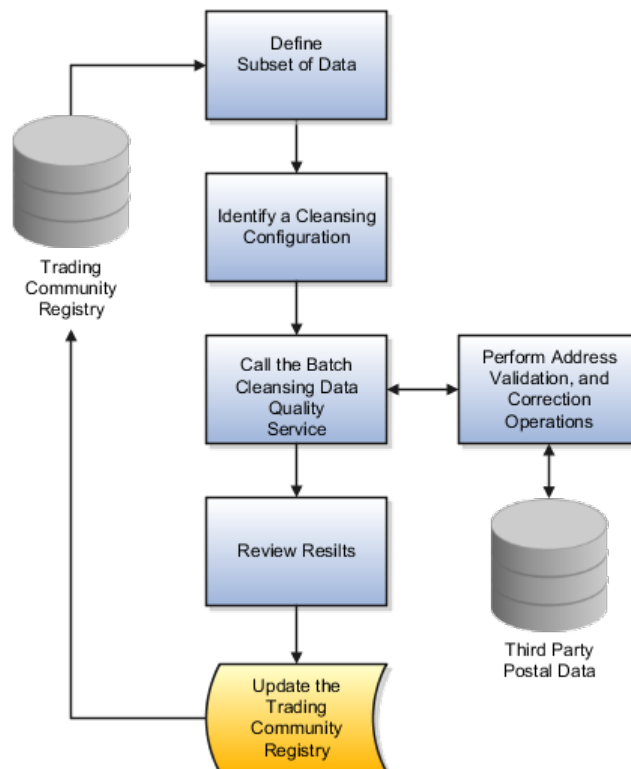
Batch Address Cleansing Components: How They Work Together

Batch cleansing enables you to perform address cleansing, standardization, and validation for a subset or entirety of the location records in the Oracle Fusion Trading Community Hub registry, or as part of a data import process.

Batch address cleansing comprises the following components:

- Batch Cleansing Service
- Data quality service consuming applications
- Cleansing configurations
- Server configurations

This figure shows the batch address cleansing components in action.



Batch Cleansing Service

The Batch Cleansing Service performs validation, standardization, and cleansing of addresses to ensure data accuracy during the import of addresses in batches, to reduce decay and obsolescence of existing address data in the trading community registry, and to resolve inherited issues from consolidated data in the registry.

Consuming Applications

Batch address cleansing finds use in master data management applications such as Oracle Fusion Trading Community Hub, and other data quality service consuming applications that import data in batches. Unlike real-time cleansing that validates addresses one by one and runs in suggest cleansing mode, batch cleansing allows for validation and correction of multiple addresses through a single call.

Cleansing Configurations

Cleansing configurations are predefined configurations used for batch and real-time cleansing of the location entity.

Oracle Fusion Trading Community Data Quality has a predefined batch cleansing configuration named Batch Data Quality Location Cleanse. The batch address cleansing service uses this cleansing configuration to derive the default country to be used to perform validation when address parsing fails to find a valid country within the address.

You can review and edit this predefined cleansing configuration. While you cannot add or delete a configuration parameter, you can change the configuration name and update the parameter values to suite your requirements.

Server Configurations

Batch address cleansing makes use of the Batch Data Quality Location Cleanse server configuration.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

Cleansing Configuration Parameters

Cleansing configuration parameters are system-level parameters that control aspects of the Oracle Fusion Trading Community Data Quality cleansing services. The Oracle Fusion Trading Community Data Quality cleansing services read values from the ZCQ_CONFIG_PARAM_VALUES and ZCQ_SO_PARAM_VALUES tables.

Cleansing configurations are predefined configurations that include parameters from both Oracle Fusion applications and the integrated third-party data quality engine. Although you cannot add or delete a configuration parameter, you can change configuration name and update the parameter values to suite your requirements. The text describes only the generic Oracle Fusion Trading

Community Data Quality parameters and their values. For third-party data quality engine specific parameters and their values, see relevant third-party documentation.

Real Time Data Quality Location Cleanse Parameters

The following parameters control real-time data cleansing operations that happen at touch point applications, such as Oracle Fusion Sales and Oracle Fusion Customer Center.

RuntimeMapping

Determines if the real-time cleansing service should use the attribute mapping defined between Oracle Fusion Trading Community Data Quality attributes and the third-party data quality engine. If the value is set to FALSE, the data quality real-time service will pass through attributes as they are between Oracle Fusion Trading Community Data Quality and the third party cleansing engine.

Batch Data Quality Location Cleanse Parameters

The following parameters control address cleansing, standardization, and validation operations for a subset or entirety of the location records in the registry, or as part of a data import process.

AMCommitRate

Controls the transaction batch size for the address cleansing process which in turn determines how often the cleansing results are committed. Set this value based on available system resources for optimum performance. If the value is too low, the data is committed often, which may cause slow performance. In contrast setting it to too high a value may require higher memory for processing.

NumChildJobs

Determines the number of child requests spawned for a batch matching request. The child requests are useful when there is a need to process data in parallel.

Manage Data Synchronization

Data Synchronization: Explained

Real-time duplicate prevention during data entry and batch duplicate identification of trading community entities in the Oracle Fusion Trading Community Hub registry and during the data import process require both initial indexing and index synchronization of the integrated third party data quality engine repository with the registry.

Periodic index synchronization of the third party data quality engine repository with the Oracle Fusion Trading Community Hub registry lets you account for the continual updates made to the registry.

Some of the entities that need to be synchronized to the matching engine repository are organization, person, and location. The synchronization should be done when these entities are created or updated.

However, no data synchronization is required for advanced batch match configuration and real-time and batch cleanse operations because of the following reasons:

- Advanced configuration is used only for batch matching such as finding duplicates within a set of interface tables records before running a bulk import process to load the records in the registry.
- Data synchronization is not needed for real-time address cleansing functions as the address is checked against the country specific postal address files and not against the existing address in the registry. Note that the postal reference files need to be updated periodically.
- During the batch cleanse operation, records in a batch are sent to the data quality server for cleansing one at a time and the data quality engine returns the cleansed record to the registry likewise, one at a time, thus completing the loop.

Synchronizing Trading Community Registry and Data Quality Engine Repository Data: Worked Example

This example demonstrates the index synchronization of the integrated third party data quality engine repository with the trading community registry. Index synchronization facilitates accounting for the changes made to the registry as part of creating new and updating existing organization, person, and location records since initial indexing. This example focuses on the index synchronization of person party records for the third-party data quality engine Informatica Identity Resolution (IIR).

Index synchronization for the Informatica Identity Resolution data quality engine involves the following tasks:

- Scheduling the synchronization process: Schedule this task to run periodically to account for the continual updates to the Oracle Fusion Trading Community Hub registry. While running the synchronization process you can select the entities for which you want to synchronize updates and specify the date and time from which to synchronize the updates. A synchronization request performs index synchronization from the last synchronized date and time until the date and time it is submitted.
- Starting Informatica Identity Resolution (IIR) Update Synchronizer: Administrators and integrators typically perform this task. Once started the synchronizer can be left running, for the life cycle of the Informatica Identity Resolution server, unless there is a need to bring it down as part of a maintenance window. However, whenever there is a loss of connectivity with the database Update Synchronizer stops functioning and needs to be manually restarted.

Note

Initial indexing and index synchronization are required only for performing matching operations aimed real-time duplicate prevention and batch duplicate

identification. Real-time and batch cleanse operations do not require initial indexing and Index synchronization.

Running the Schedule Synchronization Process

1. Navigate to Setup and Maintenance from the Tools menu.
2. Search for the Manage Data Synchronization task.
3. Click the **Go to Task** icon.
4. Select **Refresh Identity Table Information** option from the Actions menu on the Manage Data Synchronization page.
5. Enable all the relevant identity tables (for this example PER_PRIMARY_IDT, PER_ADDRESS_IDT, PER_PHONE_IDT) by selecting **Enable for Sync**, check box.
6. Enter the following Synchronization Options for each identity table:

Field	Value
Country	Leave blank when using the predefined system, FusionDQRealtime, for all countries. If using country specific system, enter the country, such as US.
Last Synchronized Time	Select Date and Time: 3/23/11 9:30:15 AM

7. Click Save.
8. Click the **Schedule Synchronization Process** button.
9. Click Advanced.
10. Select Using a Schedule. From the Frequency drop-down list, select Hourly/Minute and provide a value for the frequency. For example, 5 minutes.

Note

The frequency needs to be determined by the business requirement.

11. From the Calendar, select the End Date.

Note

The end date needs to be determined by the business requirement.

12. Submit the scheduled Sync ESS Job.
13. Click **OK** to return to the Manage Data Synchronization page.
14. Hover on the Process Status field corresponding to each relevant identity table to know the status of the child synchronization request spawned for that table.
15. Click **Save and Close**.

Starting Informatica Identity Resolution Update Synchronizer

1. Sign in the Informatica Identity Resolution (IIR) host machine.
2. Enter `cd <PROV.TOP>/InformaticaIR/bin`
3. Enter `setfusionEnv.sh`
4. Start the IIR console client using the admin option, `./idsconc -a`
5. Select **Run Synchronizer** on the **Tools** menu to launch the synchronizer.
6. In the Update Synchronizer dialog, select **All** as the value for **IDT Name**, use the default values for the rest of the fields, and click **OK**.
7. Verify that the updated and newly created person records are available in IIR, by searching for persons in the Per-dup tab of IIR Web Search Client.
8. Sign out of the Informatica Identity Resolution (IIR) user interface.

Troubleshooting Oracle Fusion Data Quality Services and IIR Servers

Troubleshooting Data Quality Services and Informatica Identity Resolution (IIR) Servers: Overview

This overview contains various troubleshooting methods pertaining to the Informatica Identity Resolution (IIR) and Data Quality (DQ), which help an administrator examine the symptoms, identify possible root causes, and take necessary steps to restore their normal functionality.

The log files (`ids.dbg`) in IIR or `iirlog` contain useful information that can help in diagnosing the root cause of the functional problems.

The troubleshooting process involves the following scenario checks:

- Matching Service
- Matching Server Administration (including Synchronizer)
- Cleansing Service
- Cleansing Server Administration

The troubleshooting steps for each scenario are grouped, and each group forms a tier. Since different root causes may share the same symptoms, the tiers are arranged in the ascending order of complexity. For effective resolution of the problem, you need to go through these tiers in the same order as presented here. Proceed to the next tier only when you are unable to confirm the root cause in the current tier.

Note

For all the scenario checks, it is assumed that IIR is running with 8 InformaticaIR processes, and that its intra-connectivity is established on the DQ Server Configuration using the correct Host and Port information.

Matching Service Troubleshooting: Explained

The Matching Service troubleshooting is spread across six tiers. Each tier indicates certain symptoms, lists possible root causes, contains the log descriptions, and prescribes the recovery procedure. While following the troubleshooting process, ensure that as part of the recovery procedure, you check for UPD locks on the Update Synchronizer before bringing up IIR.

Tier 1

Stage	Checkpoint
Symptoms	Oracle Fusion DQ Realtime Match Web services fail with a message of ZCQ_DQ_MATCH_SERVER_DISABLED or Oracle Fusion DQ Batch Match jobs fail returning a batch summary status of ERROR_STATUS_1 thus failing the Match Check Point.
Possible Root Cause	The Match Server Operation is disabled. To confirm, go to the Server Configuration page and check whether the Enabled checkbox is deselected for Realtime and Batch Basic Server.
IIR Logs	NA
Recovery	Select the Enable Matching checkbox for Realtime and Batch Basic Server.

Tier 2

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Match Services or Fusion DQ Batch Match Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Match Check Point.
Possible Root Cause	IIR is not connected. To confirm, go to DQ Matching Configuration page and check whether the Server Operation for Matching is set up with Host and Port information for IIR.
IIR Logs	NA
Recovery	Setup Server Host and Port using Manage Server Configuration FSM task.

Tier 3

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Match Services fail with an Engine Connection Error ZCQ_RT_ENG_MATCH_ERR, thus failing the Match Check Point.

Possible Root Cause	Some of the parameter values for the Matching Configuration, are incorrect. For example, the path where the IIR Server logs are written to (Verbosity Path) is modified to an invalid path by either providing a nonexistent path or the path convention is incorrect), by an Administrator. For example, d:/temp is an invalid path and thus causes failure.
IIR Logs	<pre>[SEVERE] [IirSocket] handleSSAException(> Bad rc -408040403 for ids_search_start > SSA Exception Code: (-408040403) [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] ssasrsv > It is now 20110525145629 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] ssasesi.c 5870 rc 3 14080404*100 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] ssasesi.c 2060 rc 4 140804*100 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] ssasesi.c 1571 rc 4 1408*100 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] open search 'person- dup' failed. System 'FusionDQRealtime' Rulebase 'ssa:rtunitrb' Function 'host="rws65344fwks.us.oracle.com:1668",d:/ temp' [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] apic.c 3446 rc 1 1407*1 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] closeUser: rc=-1407 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] apic.c 3340 rc 7 14*100 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] apic.c 3030 rc 14 [SEVERE] [IirSocket] SSAErrorStack: [525145629 6] Unknown Keyword 'd:/temp'</pre>
Recovery	Revert the value of the Verbosity Path field to NULL. Thus, by default the IIR would write to InformaticaIR/iirlog.

Tier 4

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Match Services or Fusion DQ Batch Match Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Match Check Point.
Possible Root Cause	<p>Any of the following:</p> <ul style="list-style-type: none"> IIR was started without setting setfusionEnv.sh or setfusionEnv.csh. This is the most common cause. DB Client on IIR was not set up correctly. For example, tnsnames.ora does not have the entry for the FUSION DB to which IIR should connect. The password for FUSION_DQ has changed or is incorrect.

IIR Logs	<pre> [518135036 ssasrv 22/ 12] errmsg_put 'SQLConnect: OCIInitialize failed 1804' [518135036 ssasrv 22/ 12] errmsg_put 'ssaoci.c 2646 rc 2' [518135036 ssasrv 22/ 12] errmsg_put 'ssa_odbc_connect: cannot connect to Data Source' [518135036 ssasrv 22/ 12] errmsg_put 'ssaodbc.c 1436 rc 8 1*100' [518135036 ssasrv 22/ 12] errmsg_put 'db_connect_user: Cannot connect to service 'rws65311' with uid 'fusion_dq' [518135036 ssasrv 22/ 12] errmsg_put 'ssaodbc.c 4112 rc 9 108*100' [518135036 ssasrv 22/ 12] errmsg_put 'ssadb8_db_opn failed' [518135036 ssasrv 22/ 12] errmsg_put 'ssaodbc.c 4441 rc 10 10809*100' [518135036 ssasrv 22/ 12] errmsg_put 'dbops.c 1561 rc 4 1080910*100' [518135036 ssasrv 22/ 12] errmsg_put 'dbldr.c 1203 rc 9 108091004*1000' [518135036 ssasrv 22/ 12] errmsg_put 'ssarbnl_rulebase_open: 'odb:0:fusion_dq/ @rws65311' db open failed -91004009' [518135036 ssasrv 22/ 12] errmsg_put 'ssarbn.c 3026 rc 7 91004009*100' [518135036 ssasrv 22/ 12] ssarbn.c(2841) ssarbnl_rulebase_destroy: rbh=0 [518135036 ssasrv 22/ 12] ssarbn.c(2909) ssarbnl_rulebase_destroy: 'odb:0:fusion_dq/ @rws65311' RULEBASE CLOSE </pre>
Recovery	<p>Follow each recovery instruction and proceed to the next step in the given order only if the current step does not fix the issue. Also, shut down IIR before performing these steps and restart it later as per instructions given for the Matching Setup.</p> <ol style="list-style-type: none"> 1. Set the environment variables using setfusionEnv.sh or setfusionEnv.csh. 2. Add correct entry to tnsnames.ora. 3. Check if the correct password is used for FUSION_DQ in InformaticaIR/ids/ssadb.dic.

Tier 5

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Match Services or Fusion DQ Batch Match Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Match Check Point

Possible Root Cause	Any of the following: <ul style="list-style-type: none">• IIR was shut down without shutting down the Admin Client (presumably running in a different session) as per the prescribed procedure. The shutdown procedure helps in detecting any active client console.• IIR was incorrectly shut down when Update Synchronizer (UPD) was running. It led to a Rule base corruption.• The DB crashed or was shut down when IIR was running.
---------------------	--

IIR Logs

```

[518143012 ssasrsv 25/ 12] errmsg_put
'ssadb8_1_file_add: FileId 3 (inuse)
already exists'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssaodbc.c 8245 rc 5'
[518143012 ssasrsv 25/ 12] errmsg_put
'dbops.c 4363 rc 3 5*100'
[518143012 ssasrsv 25/ 12] errmsg_put
'dblldr.c 3912 rc 9 503*1000'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbnl_inuse_add: Cannot create RB INUSE
table'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 2543 rc 2 503009*100'
[518143012 ssasrsv 25/ 12] errmsg_put 'IS
ANOTHER RULEBASE SERVER RUNNING?'
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase 'odb:0:fusion_dq/@rws65311'
in use by rws65311fwks.us.oracle.com
IP=130.35.116.108 on port=1668,
SSA_RB_RESTART_ID=726050355'
[518143012 ssasrsv 25/ 12] errmsg_put
'Server started at 'Wed May 18 14:21:31
2011''
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase opened at 'Wed May 18 14:23:31
2011''
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase opened as 'odb:0:fusion_dq/
@rws65311''
[518143012 ssasrsv 25/ 12] errmsg_put 'env
has SSA_RB_RESTART_ID=0'
[518143012 ssasrsv 25/ 12] errmsg_put
'Owner information: ip=130.35.116.108
pid=1236 host='rws65311fwks.us.oracle.com'
ps='260152864 /scratch/<username>/
InformaticaIR/bin/ssasrsv-n1666-m1668-x1670-
H1671-v-1/scratch/<username>/InformaticaIR/
iirlog/idssrsv.''
[518143012 ssasrsv 25/ 12] errmsg_put 'Owner
is running'
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase is locked'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbnl_inuse_create: inuse_add failed
-50300902'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 2583 rc 1'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 3031 rc 8 1*100'
[518143012 ssasrsv 25/ 12] ssarbn.c(2841)
ssarbnl_rulebase_destroy: rbh=0
[518143012 ssasrsv 25/ 12] ssarbn.c(2909)
ssarbnl_rulebase_destroy: 'odb:0:fusion_dq/
@rws65311' RULEBASE CLOSE

```

Recovery	<p>Follow each recovery instruction and proceed to the next step in the given order only if the current step does not fix the issue. Also, shut down IIR before performing these steps and restart it later as per instructions given for the Matching Setup.</p> <ol style="list-style-type: none"> 1. If the Admin Client is running, end the process or shut it down to unlock. 2. Unlock the rule base, using command line as described in Matching Setup. 3. Drop the table IDS_NN_INUSE [where nn represents the rulebase number] using the command <code>DROP IDS_00_INUSE;</code> and remove the "inuse" record from the IDS_FDT_META using the command <code>DELETE from IDS_FDT_META where Name = 'IDS_00_INUSE'.</code> 4. Create the Rule base and re-index the data afresh as described in Matching Setup. <p style="text-align: right;">Important</p> <p style="text-align: right;">Option 3 is not a supported workaround. Therefore, if it does not yield results, it is upto the Administrator to use discretion in applying Option 4.</p>
----------	--

Tier 6

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Match Services work normally, but Fusion DQ Batch Match Jobs (using Match within Batch) fail, resulting in an error.
Possible Root Cause	A typical case of DB connectivity where locks gets created on certain Rule base tables while Update Synchronizer (UPDSYNC) was still running. As a result, the Rule base is locked and cannot be accessed. The IIR Real time Services uses the Search API (which caches Rule base Information) and Scoring API (used by Match within Batch), but it fails because it cannot access the Rule base Information.

IIR Logs

```
// UPDSYNC FAILED DUE TO CONNECTION  
LOST WITH  
// DB SERVER WHILE IT WAS WRITING. THIS  
CAUSED A LOCK.  
[426200601 updsync 9/ 2] errmsg_put  
'ssaoci_exec: OCISstmtExecute failed  
-1, ORA-03135'  
[426200601 updsync 9/ 2] errmsg_put  
'ssaoci.c 3577 rc 13'  
[426200601 updsync 9/ 2] errmsg_put  
'ssadb8_sql_alloc_next_recid table:  
[01] SQLExecute ssaodbc.c 15980 failed  
SQLRET=-1 SQLSTATE=' ' NATIVE_ERR=0'  
[426200601 updsync 9/ 2] errmsg_put  
'ssadb8_sql_alloc_next_recid table:  
[02] Reason: [SSAOCI,1] OCISstmtExecute  
ssaoci.c 3561 failed 'ORA-03135:  
connection lost contact  
Process ID: 24861  
Session ID: 2108 Serial number: 50953''  
[426200601 updsync 9/ 2] errmsg_put  
'ssadb8_sql_alloc_next_recid table:  
[03] Reason: [SSAOCI,2] Stmt:  
'SELECT /*+ CHOOSE */ NEXT_RECID FROM  
IDS_FDT_RECID WHERE TFNO = :x0 FOR  
UPDATE ''  
[426200601 updsync 9/ 2] errmsg_put  
'SQLEndTran: rollback failed -1'  
[426200601 updsync 9/ 2] errmsg_put  
'ssaoci.c 3121 rc 3'  
[426200601 updsync 9/ 2] errmsg_put  
'ssadb8_sql_alloc_next_recid table:  
Error Execute ids_fdt_recid'  
[426200601 updsync 9/ 2] errmsg_put  
'ssaodbc.c 15985 rc 12 1*100'  
[426200601 updsync 9/ 2] errmsg_put  
'ssaodbc.c 16262 rc 2 112*100'  
[426200601 updsync 9/ 2] errmsg_put  
'vrec_add_sql failed'  
[426200601 updsync 9/ 2] errmsg_put  
'ssaodbc.c 10645 rc 10 11202*100'  
[426200601 updsync 9/ 2] errmsg_put  
'dbops.c 2736 rc 3 1120210*100'  
[426200601 updsync 9/ 2] errmsg_put  
'dbldr.c 2263 rc 9 112021003*1000'  
[426200601 updsync 9/ 2] errmsg_put  
'updsync.c 5971 rc 8 21003009*100'  
[426200601 updsync 9/ 2] errmsg_put  
'updsync.c 6821 rc 24 100300908*100'  
[426200601 updsync 9/ 2]  
errmsg_put 'Error processing txn  
#20110426200506222280'  
[426200601 updsync 9/ 2] errmsg_put  
'updsync.c 7039 rc 46 30090824*100'  
[426200601 updsync 9/ 2] errmsg_put  
'updsync process_txn failed -9082446'  
[426200601 updsync 9/ 2] errmsg_put  
'updsync.c 10522 rc 26 9082446*100'
```

```
// SEARCHES THROUGH SCORING API USED IN  
BATCH  
// START FAILING, SINCE THE RULEBASE  
READ FAILS  
[427125914 ssasrsv 10/ 10]  
errmsg_put 'ssadb_object_status_get:  
rulebase_read_first failed -10104021'  
[427125914 ssasrsv 10/ 10] errmsg_put  
'rbutils.c 4985 rc 2 10104021*100'  
[427125914 ssasrsv 10/ 10] errmsg_put  
'ssasesil_search_IDX_loaded: get  
status of IDX 'LOCATION_IDX' failed'  
[427125914 ssasrsv 10/ 10] errmsg_put  
'ssasesi.c 2272 rc 6 10402102*100'
```

Recovery	Shut down and start IIR after checking UPD locks and unlocking them as per the instructions in Update Synchronizer troubleshooting.
----------	---

IIR Matching Server Administration Troubleshooting: Explained

The Matching Server Administration troubleshooting is spread across four tiers. Each tier indicates certain symptoms, lists possible root causes, contains the log descriptions, and prescribes the recovery procedure. While following the troubleshooting process, ensure that as part of the recovery procedure, you check for UPD locks on the Update Synchronizer before bringing up IIR.

Tier 1

Stage	Checkpoint
Symptoms	<p>When starting the IIR Admin Console Client, it shows a pop up with a confirmation message "Rulebase doesn't exist. Do you want to create it again".</p> <hr/> <p>Note</p> <p>This popup also appears when you create a Rulebase for the first time during initial setup. However, it is an indication of problem if it appears while starting IIR Admin Console as part of regular maintenance, after the Rulebase has been created.</p>
Possible Root Cause	<p>Any of the following:</p> <ul style="list-style-type: none"> • IIR was started without setting <code>setfusionEnv.sh</code> or <code>setfusionEnv.csh</code>. This is the most common cause. • DB Client on IIR is not set up correctly. For example, <code>tnsnames.ora</code> doesn't have the entry for the FUSION DB to which IIR should connect. • The password for FUSION_DQ has changed or is incorrect.

IIR Logs	<pre>[518135036 ssasrsv 22/ 12] errmsg_put 'SQLConnect: OCIInitialize failed 1804' [518135036 ssasrsv 22/ 12] errmsg_put 'ssaoci.c 2646 rc 2' [518135036 ssasrsv 22/ 12] errmsg_put 'ssa_odbc_connect: cannot connect to Data Source' [518135036 ssasrsv 22/ 12] errmsg_put 'ssaodbc.c 1436 rc 8 1*100' [518135036 ssasrsv 22/ 12] errmsg_put 'db_connect_user: Cannot connect to service 'rws65311' with uid 'fusion_dq' [518135036 ssasrsv 22/ 12] errmsg_put 'ssaodbc.c 4112 rc 9 108*100' [518135036 ssasrsv 22/ 12] errmsg_put 'ssadb8_db_opn failed' [518135036 ssasrsv 22/ 12] errmsg_put 'ssaodbc.c 4441 rc 10 10809*100' [518135036 ssasrsv 22/ 12] errmsg_put 'dbops.c 1561 rc 4 1080910*100' [518135036 ssasrsv 22/ 12] errmsg_put 'dbldr.c 1203 rc 9 108091004*1000' [518135036 ssasrsv 22/ 12] errmsg_put 'ssarbnl_rulebase_open: 'odb:0:fusion_dq/ @rws65311' db open failed -91004009' [518135036 ssasrsv 22/ 12] errmsg_put 'ssarbn.c 3026 rc 7 91004009*100' [518135036 ssasrsv 22/ 12] ssarbn.c(2841) ssarbnl_rulebase_destroy: rbh=0 [518135036 ssasrsv 22/ 12] ssarbn.c(2909) ssarbnl_rulebase_destroy: 'odb:0:fusion_dq/ @rws65311' RULEBASE CLOSE</pre>
Recovery	<p>Following each recovery instruction and proceed to the next step in the given order only if the current step does not fix the issue. Also, shut down IIR before performing these steps and restart it later as per instructions given for the Matching Setup.</p> <ol style="list-style-type: none"> 1. Set the environment variables using setfusionEnv.sh or setfusionEnv.csh. 2. Add correct entry to tnsnames.ora. 3. Check password for FUSION_DQ in InformaticaIR/ids/ssadb.dic and make sure it is correct.

Tier 2

Stage	Checkpoint
Symptoms	<ul style="list-style-type: none"> • Starting the IIR Admin Console Client fails with a Rulebase Lock Error: "Is another Rulebase Server running?" • Starting the HTTP Search Client fails with a Rulebase Lock Error - "Is another Rulebase Server running?"

Possible Root Cause	Any of the following: <ul style="list-style-type: none"><li data-bbox="834 247 1344 390">• IIR was shut down without shutting down the Admin Client (presumably running in a different session) as per the prescribed procedure. The shutdown procedure helps in detecting any active client console.<li data-bbox="834 415 1344 499">• IIR was incorrectly shut down when Update Synchronizer (UPD) was running. It led to a Rulebase corruption.<li data-bbox="834 525 1344 579">• The DB crashed or was shut down when IIR was running.
---------------------	--

IIR Logs

```

[518143012 ssasrsv 25/ 12] errmsg_put
'ssadb8_1_file_add: FileId 3 (inuse)
already exists'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssaodbc.c 8245 rc 5'
[518143012 ssasrsv 25/ 12] errmsg_put
'dbops.c 4363 rc 3 5*100'
[518143012 ssasrsv 25/ 12] errmsg_put
'dblldr.c 3912 rc 9 503*1000'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbnl_inuse_add: Cannot create RB INUSE
table'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 2543 rc 2 503009*100'
[518143012 ssasrsv 25/ 12] errmsg_put 'IS
ANOTHER RULEBASE SERVER RUNNING?'
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase 'odb:0:fusion_dq/@rws65311'
in use by rws65311fwks.us.oracle.com
IP=130.35.116.108 on port=1668,
SSA_RB_RESTART_ID=726050355'
[518143012 ssasrsv 25/ 12] errmsg_put
'Server started at 'Wed May 18 14:21:31
2011''
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase opened at 'Wed May 18 14:23:31
2011''
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase opened as 'odb:0:fusion_dq/
@rws65311''
[518143012 ssasrsv 25/ 12] errmsg_put 'env
has SSA_RB_RESTART_ID=0'
[518143012 ssasrsv 25/ 12] errmsg_put
'Owner information: ip=130.35.116.108
pid=1236 host='rws65311fwks.us.oracle.com'
ps='260152864 /scratch/<username>/
InformaticaIR/bin/ssasrsv-n1666-m1668-x1670-
H1671-v-1/scratch/<username>/InformaticaIR/
iirlog/idssrsv.''
[518143012 ssasrsv 25/ 12] errmsg_put 'Owner
is running'
[518143012 ssasrsv 25/ 12] errmsg_put
'Rulebase is locked'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbnl_inuse_create: inuse_add failed
-50300902'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 2583 rc 1'
[518143012 ssasrsv 25/ 12] errmsg_put
'ssarbn.c 3031 rc 8 1*100'
[518143012 ssasrsv 25/ 12] ssarbn.c(2841)
ssarbnl_rulebase_destroy: rbh=0
[518143012 ssasrsv 25/ 12] ssarbn.c(2909)
ssarbnl_rulebase_destroy: 'odb:0:fusion_dq/
@rws65311' RULEBASE CLOSE

```

Recovery	<p>Follow each recovery instruction and proceed to the next step in the given order only if the current step does not fix the issue. Also, shut down IIR before performing these steps and restart it later as per instructions given for the Matching Setup.</p> <ol style="list-style-type: none"> 1. If the Admin Client is running, end the process or shut it down to unlock. 2. Unlock the rulebase, using command line as described in Matching Setup. 3. Drop the table IDS_NN_INUSE [where nn represents the rulebase number] using the command <code>DROP IDS_00_INUSE;</code> and remove the "inuse" record from the IDS_FDT_META using the command <code>DELETE from IDS_FDT_META where Name = 'IDS_00_INUSE'.</code> 4. Create the Rulebase and re-index the data afresh as described in Matching Setup. <hr/> <p style="text-align: center;">Important</p> <p>Option 3 is not a supported workaround. Therefore, if it does not yield results, it is upto the Administrator to use discretion in applying Option 4.</p>
----------	--

Tier 3

Stage	Checkpoint
Symptoms	Starting the IIR Admin Console fails with an error: "connection lost to peer".
Possible Root Cause	DB connectivity issues caused by either DB shut down, or DB crash or network connectivity issues, when IIR was running.
IIR Logs	<pre>[517153446 ssasrsv 0/ 5] 31344/xm> server started: Tue May 17 15:34:46 2011 [517153447 ssacssv 4/ 5] errmsg_put 'send(h=2, os=9) failed: Reason: Broken pipe' [517153447 ssacssv 4/ 5] errmsg_put 'socket.c 1096 rc 11' [517153447 ssacssv 4/ 5] errmsg_put 'socket.c 2559 rc 4 11*100' [517153447 ssacssv 4/ 5] errmsg_put 'ssarbcl.c 693 rc 3 1104*100' [517153447 ssacssv 4/ 5] errmsg_put 'cssys.c 2185 rc 3 502020401*100' [517153447 ssacssv 4/ 5] errmsg_put 'ssacssi.c 2125 rc 1 202040103*100' [517153448 ssacssv 4/ 5] errmsg_put 'send(h=2, os=9) failed: Reason: Broken pipe' [517153448 ssacssv 4/ 5] errmsg_put 'socket.c 1096 rc 11' [517153448 ssacssv 4/ 5] errmsg_put 'socket.c 2559 rc 4 11*100' [517153448 ssacssv 4/ 5] errmsg_put 'ssarbcl.c 7725 rc 3 1104*100'</pre>

Recovery	Close the Admin Console and restart IIR as per the Server Administration instructions.
----------	--

Tier 4

Stage	Checkpoint
Symptoms	When starting or shutting down IIR UPD Synchronizer from an active Admin Console, the Synchronizer fails with an error: "connection lost to peer".
Possible Root Cause	DB connectivity issues caused by either DB shut down, or DB crash or network connectivity issues, when IIR was running.

IIR Logs

```
[414132606 updsync 13/ 2] errmsg_put
'send(h=1, os=17) failed: Reason: Broken
pipe'
[414132606 updsync 13/ 2] errmsg_put
'socket.c 1096 rc 11'
[414132606 updsync 13/ 2] errmsg_put
'socket.c 2559 rc 4 11*100'
[414132606 updsync 13/ 2] errmsg_put
'ssarbcl.c 7725 rc 3 1104*100'
[414132606 updsync 13/ 2] errmsg_put
'SQLEndTran: commit failed -1'
[414132606 updsync 13/ 2] errmsg_put
'ssaoci.c 3128 rc 4'
[414132606 updsync 13/ 2] errmsg_put
'ssadb8_db_commit: [01] SQLEndTran
ssaodbc.c 19388 failed SQLRET=-1 SQLSTATE='
' NATIVE_ERR=0'
[414132606 updsync 13/ 2] errmsg_put
'ssadb8_db_commit: [02] Reason: [SSAOCI,1]
OCITransCommit ssaoci.c 3124 failed
'ORA-03114: not connected to ORACLE''
[414132606 updsync 13/ 2] errmsg_put
'SQLEndTran: rollback failed -1'
[414132606 updsync 13/ 2] errmsg_put
'ssaoci.c 3121 rc 3'
[414132606 updsync 13/ 2] errmsg_put
'ssadb8_db_commit: Commit failed'
[414132606 updsync 13/ 2] errmsg_put
'ssaodbc.c 19390 rc 1 1*100'
[414132606 updsync 13/ 2] errmsg_put
'ssaodbc.c 5065 rc 14 101*100'
[414132606 updsync 13/ 2] errmsg_put
'dbops.c 1605 rc 3 10114*100'
[414132606 updsync 13/ 2] errmsg_put
'dblldr.c 1253 rc 9 1011403*1000'
[414132606 updsync 13/ 2] errmsg_put
'db_close(@2420) failed -11403009'
[414132606 updsync 13/ 2] errmsg_put
'utils.c 1739 rc 1'
[414132606 updsync 13/ 2] errmsg_put
'utils.c 2420 rc 7 1*100'
[414132607 updsync 13/ 2] errmsg_put
'send(h=1, os=17) failed: Reason: Broken
pipe'
[414132607 updsync 13/ 2] errmsg_put
'socket.c 1096 rc 11'
[414132607 updsync 13/ 2] errmsg_put
'socket.c 2559 rc 4 11*100'
[414132607 updsync 13/ 2] errmsg_put
'ssarbcl.c 2248 rc 6 1104*100'
[414132608 updsync 13/ 2] errmsg_put
'send(h=1, os=17) failed: Reason: Broken
pipe'
[414132608 updsync 13/ 2] errmsg_put
'socket.c 1096 rc 11'
[414132608 updsync 13/ 2] errmsg_put
'socket.c 2559 rc 4 11*100'
[414132608 updsync 13/ 2] errmsg_put
'ssarbcl.c 693 rc 3 1104*100'
[414132609 updsync 13/ 2] errmsg_put
'send(h=1, os=17) failed: Reason: Broken
pipe'
[414132609 updsync 13/ 2] errmsg_put
'socket.c 1096 rc 11'
[414132609 updsync 13/ 2] errmsg_put
'socket.c 2559 rc 4 11*100'
[414132609 updsync 13/ 2] errmsg_put
'ssarbcl.c 7725 rc 3 1104*100'
[414132610 updsync 13/ 2] errmsg_put
'send(h=1, os=17) failed: Reason: Broken
pipe'
[414132610 updsync 13/ 2] errmsg_put
'socket.c 1096 rc 11'
[414132610 updsync 13/ 2] errmsg_put
'socket.c 2559 rc 4 11*100'
```

Recovery	Close the Admin Console and restart IIR as per the Server Administration instructions.
----------	--

Update Synchronizer Troubleshooting: Explained

The root cause of Update Synchronizer (UPD) locks is either an incorrect shut down where the Search Server was shut down while UPD was still running followed by ending the UPD process or an abrupt loss in DB connectivity when UPD was processing rows to be synchronized. To restore normal functioning of IIR, you need to delete the UPD locks.

To detect UPD locks, if there are any, run the command `./lockmgr list uq -rssa:<rulebase alias> -h<IIR RB Host>:<IIR RB Port> -l in <PROV.TOP>/InformaticaIR/bin`. For example, if you run the following command:

```
./lockmgr list uq -rssa:rtunitrb -haufsn4ayf04:1668 -l
```

it returns the following details:

```
sorted locks:
uq: updsync:1302896463:6826
  obj: idtsyncsystem
  id: system=fusiondqrealtime
  type: w
  lock: ip=144.23.252.42 pid=31928
host='aufsn4agf04.oracleoutsourcing.com' ps='80154501
/u01/APPLTOP/fusionapps/InformaticaIR/bin/updsync-t60-m100-
aufsn4agf04.oracle
outsourcing.com:1668-rssa:rtunitrb-vus-pf'
```

Note

UPD Locks should be deleted only when you find them, even when UPD has been shut down and you do not have any UPDSYNC process running.

To delete the locks, run the command `./lockmgr del uq -rssa:<rulebase alias> -h<IIR RB Host>:<IIR RB Port> in <PROV.TOP>/InformaticaIR/bin`. For example,

```
./lockmgr del updsync:1302896463:6826 -rssa:rtunitrb -haufsn4ayf04:1668
```

Cleansing Service Troubleshooting: Explained

The Cleansing Service troubleshooting is spread across five tiers. Each tier indicates certain symptoms, lists possible root causes, contains the log descriptions, and prescribes the recovery procedure.

Tier 1

Stage	Checkpoint
Symptoms	Fusion DQ Realtime Cleanse Web Services fail with a message of <code>ZCQ_CLEANSE_SERVER_DISABLED</code> or Fusion DQ Batch Match Jobs fail returning a Batch Summary Status of <code>ERROR_STATUS_1</code> thus failing the Cleanse Check Point.
Possible Root Cause	The Cleanse Server Operation is disabled. To confirm, go to the Edit Server Configuration page and check whether the Enable Cleansing checkbox is deselected for Realtime Cleanse Server and Batch Cleanse Server.
IIR Logs	NA
Recovery	Select the Enable Cleansing checkbox for Realtime Cleanse and Batch Cleanse Servers.

Tier 2

Stage	Checkpoint
Symptoms	Fusion DQ Cleansing Services or Fusion DQ Batch Cleanse Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Cleansing Check Point.
Possible Root Cause	IIR connection setup is missing or incorrect in the cleansing configuration. Using the Manage Server Configuration FSM task, check the Server Host and Port information for the cleansing servers.
IIR Logs	NA
Recovery	Setup IIR Host and Port in Cleansing Configuration page

Tier 3

Stage	Checkpoint
Symptoms	Fusion DQ Cleansing Services or Fusion DQ Batch Cleanse Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Cleansing Check Point.
Possible Root Cause	ASM Key stored in InformaticaIR/ssaas/ad5/ad/db has expired.
IIR Logs	<pre>[523163408 ssasrsv 14/ 12] errmsg_put 'as_init: Object init failed: (-1601) Critical Error: The engine usage period has expired' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1338 rc 4' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1390 rc 1 4*100' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 3097 rc 2 401*100' [523163408 ssasrsv 14/ 12] errmsg_put 'ssasesi_addr_init: Error -40102 initializing Addr Std module' [523163408 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13477 rc 9 40102*100'</pre>
Recovery	Get the correct ASM key from Address Doctor and replace the existing key in the file called key in InformaticaIR/ssaas/ad5/ad/db. Subsequently, restart the IIR Server.

Tier 4

Stage	Checkpoint
Symptoms	Fusion DQ Cleansing Services or Fusion DQ Batch Cleanse Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Cleansing Check Point.
Possible Root Cause	ASM Key stored in InformaticaIR/ssaas/ad5/ad/db is either incorrect or corrupt.

IIR Logs	<pre>// junk or error ASM key error [523163604 ssasrsv 14/ 12] errmsg_put 'as_init: Object init failed: (-1600) Critical Error: No valid unlock code was given' [523163604 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1338 rc 4' [523163604 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1390 rc 1 4*100' [523163605 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 3097 rc 2 401*100' [523163605 ssasrsv 14/ 12] errmsg_put 'ssasesi_addr_init: Error -40102 initializing Addr Std module' [523163605 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13477 rc 9 40102*100'</pre>
Recovery	Get the correct ASM key from Address Doctor and replace the existing key in the file called key in InformaticaIR/ssaas/ad5/ad/db. Subsequently, restart the IIR Server.

Tier 5

Stage	Checkpoint
Symptoms	Fusion DQ Cleansing Services or Fusion DQ Batch Cleanse Jobs fail with an Engine Connection Error ZCQ_CMN_ENG_CONNECT_ERR, thus failing the Cleansing Check Point.
Possible Root Cause	ASM Key stored in InformaticaIR/ssaas/ad5/ad/db does not exist.
IIR Logs	<pre>[523163811 ssasrsv 14/ 12] errmsg_put 'fopen('/slot/ems4517/appmgr/InformaticaIR/ ssaas/ad5/ad/db/key', 'r') failed 16: No such file or directory' [523163811 ssasrsv 14/ 12] errmsg_put 'fopen.c 985 rc 19' [523163811 ssasrsv 14/ 12] errmsg_put 'fopen.c 1070 rc 3 19*100' [523163811 ssasrsv 14/ 12] errmsg_put 'fopen.c 1141 rc 2' [523163811 ssasrsv 14/ 12] errmsg_put 'ssasesi_addr_get_key_and_workdir: open Addr Std license key file '/slot/ems4517/ appmgr/InformaticaIR/ssaas/ad5/ad/db/key' failed' [523163811 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13293 rc 6' [523163811 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13443 rc 6 6*100'</pre>
Recovery	Get the correct ASM key from Address Doctor and create a file called key in InformaticaIR/ssaas/ad5/ad/db. Subsequently, restart the IIR Server.

IIR Cleansing Server Administration Troubleshooting: Explained

The Cleansing Server Administration troubleshooting process indicates certain symptoms, lists possible root causes, contains the log descriptions, and prescribes the recovery procedure.

Stage	Checkpoint
Symptoms	The IIR Cleansing Console does not respond to the start command <code>asmcli -b</code> run from <code>InformaticaIR/bin</code> , thus failing the Cleansing Setup Checkpoint.
Possible Root Cause	ASM Key stored in <code>InformaticaIR/ssaas/ad5/ad/db</code> has either expired, is either incorrect or corrupt, or does not exist.
IIR Logs	<ul style="list-style-type: none"> <pre>[523163408 ssasrsv 14/ 12] errmsg_put 'as_init: Object init failed: (-1601) Critical Error: The engine usage period has expired' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1338 rc 4' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1390 rc 1 4*100' [523163408 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 3097 rc 2 401*100' [523163408 ssasrsv 14/ 12] errmsg_put 'ssasesi_addr_init: Error -40102 initializing Addr Std module' [523163408 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13477 rc 9 40102*100'</pre> <pre>[523163604 ssasrsv 14/ 12] errmsg_put 'as_init: Object init failed: (-1600) Critical Error: No valid unlock code was given' [523163604 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1338 rc 4' [523163604 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 1390 rc 1 4*100' [523163605 ssasrsv 14/ 12] errmsg_put 'ssaas5.c 3097 rc 2 401*100' [523163605 ssasrsv 14/ 12] errmsg_put 'ssasesi_addr_init: Error -40102 initializing Addr Std module' [523163605 ssasrsv 14/ 12] errmsg_put 'ssasesi.c 13477 rc 9 40102*100'</pre>
Recovery	Get the correct ASM key from Address Doctor and replace the existing key in the file called <code>key</code> in <code>InformaticaIR/ssaas/ad5/ad/db</code> . Subsequently, restart the IIR Server.

FAQs for Define Data Quality

What's the difference between different matching configurations and matching server configurations?

Matching configurations are used for real-time and batch matching of party entities with a view to preventing and identifying duplicate entries in the trading community registry. The matching configuration includes the parameters that can be set at the matching configuration level and can be modified depending on matching strategy, data, and result requirements. Use the matching configuration to view or change the values of the matching configuration level parameters.

Matching server configurations provide the address and port of the third party server to which the match request should be sent for processing. Server

configurations show all the parameters that can be set at either the matching configuration level or the server configuration level along with the parameter type (third party data quality engine or Oracle Fusion Trading Community Data Quality) and the cardinality of the parameter. The parameters that are set at the server level are applicable to all the matching configurations. Use the server configuration to view or change the values of the server configuration level parameters.

What's the difference between different cleansing configurations and cleansing server configurations?

Cleansing configurations are used for real-time and batch cleansing of addresses during data entry and on a periodical basis to cleanse, standardize, and validate existing addresses to ensure data accuracy. The cleansing configuration includes the parameters that can be set at the cleansing configuration level and can be modified depending on cleansing strategy, data, and result requirements.

Cleansing server configurations provide the address and port of the third party data quality server to which the cleanse request should be sent for processing. There are no server configurations level parameters for cleansing.

What's the difference between real-time duplicate prevention and duplicate identification of trading community entities?

Real-time duplicate prevention involves identification of all possible duplicate records that may exist in the trading community registry for an entered record. This enables prevention of individual entry of duplicate trading community entities, such as organization, person, or location, through calling touch point applications, such as Oracle Fusion Sales and Oracle Fusion Customer Center in the registry.

Duplicate identification of trading community entities is achieved by batch matching. The objective is to identify potential duplicate entities already existing in the trading community registry, and then resolve the duplicates by merging or linking.

What's the difference between basic duplicate identification and advanced duplicate identification of trading community entities?

The objective of basic duplicate identification is to resolve the duplicate entities in the Oracle Fusion Trading Community Hub registry by merging or linking.

Advanced duplicate identification is mainly done to find duplicates in a group of records before uploading them to the Oracle Fusion Trading Community Hub registry. For example, an acquiring company may do an advanced duplicate identification of the legacy data of the acquired company before uploading the data to the registry.

What's the difference between real-time address cleansing and batch address cleansing?

Real-time address cleansing is an online, interactive service to cleanse, standardize, and validate addresses during the data entry process.

Batch address cleansing enables you to cleanse, standardize, and validate addresses that are either existing in the Oracle Fusion Trading Community Hub registry or are being imported into it.

Define Customer Hub Configuration

Run Request Dispatch Job

Configuring the Data Quality Management Request Dispatcher: Explained

Use the request dispatcher to process cleansing and duplicate identification batches, and duplicate resolution requests. The Run Request Dispatcher process can be scheduled to run as one time process or at a repeat schedule. This task can also be used to monitor the cleansing and duplicate identification batches and duplicate resolution requests.

You can configure the request dispatcher to run in the following two modes:

- Basic
- Advanced

Running the Dispatcher in the Basic Mode

Run the dispatcher in this mode to submit the request dispatcher job for immediate processing of the duplicate identification batches, cleansing batches, or resolution requests

Running the Dispatcher In the Advanced Mode

This mode lets you schedule the request dispatcher job to be run either immediately or at a specified intervals, such as, every 15 minutes to process duplicate identification batches, cleansing batches, or resolution requests.

Manage Customer Hub Profile Options

Customer Hub Profile Options: Explained

The Customer Hub profile options provide data access and processing options related to data governance, duplicate identification processes, data cleansing processes, and duplicate resolution requests.

Note

The Hub profile option values can be set only at the site level using the predefined profile option definitions.

Navigate to Manage Customer Hub Profile Options page using the Setup and Maintenance option on the Tools menu to configure the following profile options:

- Address Cleansing Configuration
- Data Cleansing Process Batch Size
- Resolution Request Type Default
- Duplicate Identification Organization Configuration
- Duplicate Identification Person Configuration
- Duplicate Identification Process Batch Size
- User Merge Requests

Address Cleansing Configuration

Specify the data quality address cleansing configuration used by the data cleansing process.

Default value: BT_LOC_CLEANSE

Data Cleansing Process Batch Size

Control the transaction batch size for the data cleansing process. Set this value based on available system resources. Set this value based on available system resources.

Default value: 100

Resolution Request Type Default

Specify the default request type value for duplicate resolution requests.

Default value: Merge

Duplicate Identification Organization Configuration

Specify the data quality organization match configuration used by the duplicate identification process.

Default value: BT_ORG_DUP_BASIC

Duplicate Identification Person Configuration

Specify the data quality person match configuration used by the duplicate identification process.

Default value: BT_PERSON_DUP_BASIC

Duplicate Identification Process Batch Size

Control the transaction batch size for the duplicate identification process. The value is used to chunk records in the batch and process each chunk in a loop as a separate transaction.

Default value: 100

User Merge Requests

Specify the processing of merge requests, such as, process without approval and approval needed. The different values of this profile option are explained below.

- **Allow Processing Without Approval:** If the User Merge Requests profile option is set to this value, user merge requests are processed immediately without approval by Data Steward.
- **Process Subject to Approval:** If the User Merge Requests profile option is set to this value, user merge requests are entered for review by Data Steward, who can decide to process them or not.
- **Unspecified (NULL):** This default values of the User Merge Requests profile option indicates that the Customer Hub is not configured and that user merge requests can be processed immediately without the need for Data Steward approval.

Note

The default value of the User Merge Requests profile option , Unspecified or NULL, will enable the user to submit the merge request to be processed immediately without the need for Data Steward approval only if the user has the Submit Trading Community Merge Request privilege in addition to the Enter Trading Community Merge Request privilege.

Sales accounts merges initiated from the Customer Center, or the Enter Merge Request setup task, or the Automerge web service, `DuplicateResolutionRequestService`, cannot happen immediately without Data Steward processing. Note that to allow Data Steward processing the User Merge Requests Profile Option should be set to Process Subject to Approval , otherwise the merge requests will error.

Manage Survivorship Rules

Survivorship Rules: Explained

Survivorship rules are a collection of business rules defined for selecting master record and attributes to be retained during merge or link, or update operations.

Survivorship rules facilitate intelligent creation of the best version record from multiple source systems, based on business rules. You can configure

survivorship rules to determine how conflicts should be resolved while merging or updating duplicate records from different source systems to create the best version record. You can specify criteria to determine the master (or the surviving) party record for duplicate resolution. The survivorship rules criteria determine the attributes of the best version record that should be retained from a particular source system. The rules use the priority assigned to different source systems, as well as other criteria to determine which values to retain for certain attributes.

Managing Survivorship Rules

Use the UI available from the Manage Survivorship Rules Setup and Maintenance task to view, create, edit, and delete survivorship rules.

The survivorship functionality is disabled by default. To enable survivorship, set the `ZCH_ENABLE_SURVIVORSHIP` profile option to Yes using the Manage Customer Hub Profile Options Setup and Maintenance task.

Importing and Exporting Survivorship Rules

Use the Import Survivorship Rules Dictionary web service to import the survivorship rules dictionary XML file created or stored elsewhere into the Oracle Fusion Applications environment. Use the Export Survivorship Rules Dictionary web service to export survivorship rules dictionary from the Oracle Fusion Applications environment.

Defining Survivorship Rules: Worked Example

This example demonstrates how to create a survivorship rule. Survivorship rules enable intelligent creation of the best version record, especially from multiple source systems, by specifying criteria for selecting the record to be retained during a merge operation.

Use the UI available from the Manage Survivorship Rules Setup and Maintenance task to create a survivorship rule and to specify the criteria for selecting the master record.

Create Survivorship Rule

1. Navigate to the Manage Survivorship Rules page.
2. Click Create.
3. Enter the following sample information.

Field	Value
Rule Name	PickPersonMasterRule
Description	Select the master person record based on original source system of the record.
Rule Type	Set_Master

Object Type	Person
-------------	--------

Note

You can create a survivorship rule for the following two types of party records: person and organization.

4. Click Apply. You are taken to the Define Survivorship Rules: Select Master Record page.

Specify Criteria for Selecting the Master Record

1. On the Define Survivorship Rules: Select Master Record page, enter the following IF/THEN rules condition.

Rule Condition	Value
IF Condition	<pre>IF PersonParty is a TCHFactTypeDictionary.PersonPartyVO and there is a case where { OrigSourceSystem != null and OrigSourceSystem.OwnerTableId == PersonParty.PartyId and OrigSourceSystem.OrigSystem == "GSI" }</pre>
THEN Condition	<pre>THEN Assert new Result (name:"masterId", value:"PersonParty.PartyId)</pre>

Note

If survivorship rules conflict with a real life situation, an error message is displayed and the merge fails. In the context of this example, the survivorship rule in use states that if a person party record is from the source system GSI, it should be set as the master record. Consequently, if at runtime a merge request has more than one person record from GSI, the survivorship engine returns multiple master records, which is incorrect. In such a situation, an error message is displayed stating that the merge request failed because conflicting survivorship rules are defined and that the administrator should modify the rules to ensure that only one master record is returned.

2. Verify that the effective date of the rule is set to the rule's creation date and, if required, modify it as appropriate.
3. Select Medium as the priority for the rule.
4. Select the Rules Active option.
5. Click Save or Save and Close.
6. Click Submit.

FAQs for Manage Survivorship Rules

What's the difference between survivorship rules and merge agreement rules?

Survivorship rules are a collection of business rules defined for the creation of the best version record intelligently, especially from multiple source systems, by specifying criteria for selecting record and attributes to be retained during merge or link, or update operations.

A merge agreement rule is a collection of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Merge requests that violate these rules are either rejected or end in error.

Manage Party Center Trees

Party Center: Explained

The party center functionality enables a comprehensive management of party information. A party is a physical entity, such as a person, organization or group, that the deploying company has an interest in tracking. The party center functionality lets you bring a particular party into context, fetch data from various systems about the party in context, and review and edit the data in one location.

The party center enables you to do the following:

- Create Organizations
- Manage Organizations
- Create Groups
- Manage Groups
- Create Persons
- Manage Persons

Party Center Tree

You can select a party from the organizations, person, or groups lists to bring it into context. Once a party is in context, you can fetch, review, and edit its data in the party center. The party entities are displayed as nodes in the party center tree. A party center tree comprises nodes that let you fetch data related to the party in context from various systems, and review and edit the data displayed in each node.

The party center nodes display information, such as profile, profile history, party usage assignments, relationships, classifications, source system references, and linked trading community members, about the party in context.

Party Center Trees: Explained

A party center tree comprises nodes that let you fetch data about the party in context from various systems, and review and edit the data displayed in each node. You can also customize a party center tree by showing or hiding its various nodes as required, reviewing and editing the node names as appropriate, and determining the need for entering additional parameter values to gain access to a node.

There are three types of party center trees:

- **Organization:** An organization tree displays nodes for organization parties modelled in the trading community foundation across all usages.
- **Group:** A group tree displays nodes for group parties modelled in the trading community foundation across all usages.
- **Person:** A person tree displays nodes for person parties modelled in the trading community foundation across all usages. All nodes on the person tree are visible to all users.

Each tree displays slightly different nodes, and the information that you can view and edit on each node depends upon your security privileges.

A party center tree may include the following nodes:

- Profile
- Profile History
- Usage assignments
- Relationships
- Classifications
- Source system references
- Linked parties

Note

The profile history information for a group party type is displayed in the Profile node.

Some of them are explained below:

Profile

This node lets you review and edit party profile information such as name, party usage, additional names, additional identifiers, addresses, and contact points.

Profile History

This node lets you view the history of changes to profile records of the party, person or organization, in context in the party center.

Usage Assignments

This node launches a task flow that gives visibility into what party usages have been currently assigned to the party. Party usages that allow manual create, manual update, and unconditional assignment are maintainable through this user interface.

Relationships

This node lets you view, create, edit, and delete all relationships for the party in context in the Party Center. The node also displays the contact details of a relationship.

Classifications

This node lets you view, create, edit, and delete classifications for the party in context in the Party Center. The displays the primary classification first in the list of classifications.

Source System References

This node lets you view, create, edit, and delete all source system references contributing to the master organization party record in context in the Party Center.

Linked Parties

This node lets you manage the parties linked to the party in context.

Managing Party Center Trees: Explained

Use the Setup and Maintenance option on the Tools menu to navigate to the Define Customer Hub Configuration tasklist. Go to the Manage Party Center Tree task to manage a party center tree. Alternatively, use the Actions menu options in the regional area of the Oracle Fusion Trading Community Hub to personalize a party center tree.

Specify the following attributes for each node in the party center tree:

- Name
- Visible
- Parameter

Name

Set this attribute to specify the name of the node that you want to display in the party center tree.

Visible

Set this attribute to specify the display status, hidden or visible, of a node in the party center tree.

Parameter

Specify the additional parameter values required to gain access to a node.

Manage Customer Data Hub Notes

Defining Notes: Points to Consider

A note is a record attached to a business object that is used to capture nonstandard information received while conducting business. When setting up notes for your application, you should consider the following points:

- Note Types
- Note Type Mappings

Note Types

Note types are assigned to notes at creation to categorize them for future reference. During setup you can add new note types, and you can restrict them by business object type through the process of note type mapping.

Note Type Mappings

After note types are added, you must map them to the business objects applicable to your product area. Select a business object other than Default Note Types. You will see the note types only applicable to that object. If the list is empty, note type mapping doesn't exist for that object, and default note types will be used. Select Default Note Types to view the default note types in the system. Modifying default note types will affect all business objects without a note type mapping. For example, you have decided to add a new note type of Analysis for your product area of Sales-Opportunity Management. Use the note type mapping functionality to map Analysis to the Opportunity business object. This will result in the Analysis note type being an available option when you are creating or editing a note for an opportunity. When deciding which note types to map to the business objects in your area, consider the same issues you considered when deciding to add new note types. Decide how you would like users to be able to search for, filter, and report on those notes.

Note

Extensibility features are available on the Note object. For more information refer to the article [Extending CRM Applications: how it works](#).

Manage Agreement Rules

Agreement Rules: Explained

An agreement rule is a collection of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Merge requests that violate these rules are either rejected or end in error.

Agreement rules let you check a merge request for any veto conditions that can prevent a merge from occurring. These rules save resources and time by obviating the need to review merge request to prevent undesired merge from being processed. Besides, agreement rules prompt you to consider alternative duplicate resolution mechanism such as linking.

Agreement rule can be of the following two types:

- Predefined
- User-defined

Predefined Agreement Rules

These are agreement rules that are predefined in the application. You can only view the predefined agreement rules.

The following table describes the predefined agreement rules for the Oracle Fusion Trading Community Hub. Merge requests that violate these rules are automatically rejected.

Name	Description
HR_APPLICANT_VETO	Prevents a person party with an active party usage of HR_APPLICANT from merging with another party.
HR_EMPLOYEE_VETO	Prevents a person party with an active party usage of HR_EMPLOYEE from merging with another party.
HR_CONTINGENT_WORKER_VETO	Prevents a person party with an active party usage of HR_CONTINGENT_WORKER from merging with another party.
HR_PARTY_SITE_VETO	Prevents a party site with active original system reference from Fusion HCM system from merging with another party site.
RESOURCE_PERSON_VETO	Prevents a person party with an active party usage of RESOURCE from merging with another party.
BANK_VETO	Prevents an organization party with an active party usage of BANK from merging with another party.
CLEARINGHOUSE_VETO	Prevents an organization party with an active party usage of CLEARINGHOUSE from merging with another party.

BANK_BRANCH_VETO	Prevents an organization party with an active party usage of BANK_BRANCH from merging with another party.
BRANCH_CLEARINGHOUSE_VETO	Prevents an organization party with an active party usage of CLEARINGHOUSE_BRANCH from merging with another party.
LEGAL_EXTLEGAL_PARTYUSGRULE_VETO	Prevents an organization party with active party usage of LEGAL_ENTITY from merging with another organization party with active party usage of EXTERNAL_LEGAL_ENTITY.
IC_PARTICIPANT_PERSON_VETO	Prevents a person party with an active party usage of INCENTIVE_COMP_PARTICIPANT from merging with another party.
IC_PARTICIPANT_ORG_VETO	Prevents an organization party with an active party usage of INCENTIVE_COMP_PARTICIPANT from merging with another party.
PARTNER_VETO	Prevents an organization party with an active party usage of PARTNER from merging with another party.
INACTIVE_PARTNER_VETO	Prevents an organization party with an active party usage of INACTIVE_PARTNER from merging with another party.
CUST_CONTACT_DIFF_RESOURCE_ORG_VETO	Prevents two partner owned contacts belonging to different resource organizations from being merged.
CUST_CONTACT_INTERNAL_PARTNER_VETO	Prevents a partner owned contact that does not belong to a resource organization from merging with another partner owned contact that belongs to a resource organization.
CUST_CONTACT_INTERNAL_PARTNER_ORG_VETO	Prevents a contact owned by an internal user from merging with a contact owned by a partner user.
CARD_ISSUER_VETO	Prevents an organization party with an active party usage of CREDIT_CARD_PROVIDER from merging with another party.
LEGAL_ENTITY_VETO	Prevents an organization party with an active party usage of LEGAL_ENTITY from merging with another party.
ESTABLISHMENT_VETO	Prevents an organization party with an active party usage of ESTABLISHMENT from merging with another party.
HIERARCHY_CYCLE_PREVENTATION_VETO	Prevents merge from happening when the master party record is at a level lower than the nonmaster party record within the same active tree version.
NAMEDACCOUNT_UNNAMEDACCOUNT_VETO	Prevents merge from happening when the master party record is not a named account whereas the nonmaster party record is a named account and these are in active hierarchies.

User-defined Agreement Rules

These are additional agreement rules that you can define to determine whether a merge request should be vetoed by the application. You can create, view, update, and delete user-defined agreement rules.

Defining Agreement Rules: Worked Example

This example demonstrates how to create user-defined agreement rules that you can use to prevent a merge request from being processed.

Agreement rules are collections of patterns and conditions that are defined to determine whether a merge request should be vetoed by the application or not. Perform the following tasks to define agreement rules:

- Select the relevant agreement rules dictionary
- Add a new agreement rule

For more information on agreement rules, see Oracle Fusion Middleware User's Guide for Oracle Business Rules on Oracle Technology Network at <http://www.oracle.com/technetwork>.

Select Agreement Rules Dictionary

An agreement rules dictionary is a collection of predefined terms and attributes for which agreement rules can be defined. Use the following steps to select the relevant agreement rules dictionary:

1. Navigate to Manage Agreement Rules: Define Dictionary page as follows:
Set Up Tasks - Manage Agreement Rules - Define Dictionary
2. Select the relevant agreement rules dictionary from the **Select Dictionary** drop down list and click **Load** to load its predefined terms.
3. Select **Visible** for the terms and term attributes that you want to be available when defining rules later on in the flow.
4. Click Save or Save and Close.

Add a New Agreement Rule

1. Navigate to Manage Agreement Rules: Define Rules page.
2. Add a new rule and provide a rule name.
3. Click on Define Pattern.
4. In the **Justification Reason**, enter the reason for creating the agreement rule.
5. In the **Justification Area** enter the scope of the agreement rule.
6. Add a new pattern and complete the fields, using the sample information provided in this table. Use the default values except where indicated. Note that the relation is always AND between patterns and cannot be edited.

Pattern	Dictionary Term	Term Alias	Relation
for each case where	PersonPartyVO	Person	AND
for each case where	OrganizationParty\	NonmasterParty	AND

there is a case where	PartyUsageAssignm	PartyUsageAssignm	AND
-----------------------	-------------------	-------------------	-----

- To define term conditions for each pattern defined, select the pattern in the Patterns table and navigate to the Conditions table underneath.
- Add a new condition and complete the fields using the sample information provided in this table. Use the default values except where indicated.

Term Attribute	Operator	Value	Relation
Person.PartyNumber - missing PartyNumber	is not	1234	AND
NonmasterParty.Me	=	Nonmaster	AND
UsageAssignment.I	=	HR_APPLICANT	AND

- Click Save or Save and Close.
- Click Submit.

Enter Merge Request

Manual Merge: Explained

Manual merge involves selection of duplicate records directly by the user of a merge functionality consuming application. Examples of such applications include Oracle Fusion Sales and Receivables.

Manual merge is unlike the system merge in which duplicates are selected automatically using a matching configuration. Another difference is that while only two records can be merged as part of manual merge multiple records can be merged as part of system merge.

Manually merge the two records that you think are duplicates as in one of the following two ways:

- Immediate manual merge
- Deferred manual merge

Immediate Manual Merge

Involves specification of one of the two duplicate records as the master record for an immediate merge. In an immediate merge, while you can specify some of the attributes from individual records to be mapped to the master record, other attributes are automatically transferred.

Deferred Manual Merge

Involves submission of merge request to be reviewed and processed later by a data steward.

Merging Records Manually: Worked Example

This example demonstrates how the business user of a merge functionality consuming application, such as Oracle Fusion Sales or Receivables, can merge manually two duplicate records. This example illustrates only the generic process for merging records manually from the Enter Merge Request task in the Define Customer Hub Configuration central setup task list. Alternatively, as the business user of a merge functionality consuming application, you can merge two records that you think are duplicate from your own application UI pages that you use to manage customer information.

Merging records manually involves the following tasks:

- Identifying duplicate records
- Selecting the master and duplicate records
- Submitting the merge

Identifying Duplicate Records

1. Navigate to the Search and Select: Record to Merge page as follows: **Set Up Tasks Enter Merge Request Search and Select: Record to Merge**.
2. To search and select the record to be merged, complete the search criteria using the sample information provided in the given table. Use the default values except where indicated.

Name	Registry ID (Optional)
John	15918

3. Click Search.
4. From the search results, select the first record to be merged.
5. Select **Merge Record** from the **Actions** menu

Selecting the Master and Duplicate Records

Note

The function privilege Enter Trading Community Merge Request controls whether the **Merge** secondary window appears at all.

1. In the **Merge** secondary window, specify the selected record either as master or duplicate.
2. To select the second (master or duplicate) record to be merged, in the **Party Picker** enter the sample information provided in the given table. Use the default values except where indicated.

Name	Registry ID (Optional)
John	10854

3. Click Search.

Submitting the Merge

1. On the Map Profile Attributes page, specify the attributes from each of the two records, such as name and gender, that should be mapped to the master record. Any child entities, such as addresses, accounts, account addresses, relationships, and contact points, are transferred to the master record.
2. Click **Submit**. The merge happens immediately if the data governance profile: User merge requests is set to Allow Processing Without Approval. If the profile is set to Process Subject to Approval, the merge is reviewed and processed later by a data steward.

Define Extensions for Customer Data Management

Define Custom Enterprise Scheduler Jobs

Managing Job Definitions: Highlights

Oracle Enterprise Scheduler jobs are run in Oracle Fusion Applications to process data and, in some cases, to provide report output. A job definition contains the metadata that determines what the job does and what options are available to users who run the job. You can create and maintain job definitions for use in Oracle Fusion Applications.

Managing job definitions is fully described in the Oracle Fusion Applications Administrator's Guide. As you read content from that guide, note that the guide describes managing Oracle Enterprise Scheduler, including job definitions, from Oracle Enterprise Manager Fusion Applications Control. You can also access job definitions by starting in the Setup and Maintenance Overview page and searching for the Enterprise Scheduler job tasks for your applications.

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 and Creating Job Definitions

- You can access predefined and custom job definitions.

See: Viewing Job Definitions

- You can create jobs based on Java, PL/SQL, or any other supported technology.

See: Creating a Job Definition

- If you are using the Setup and Maintenance work area, then the Enable submission from Enterprise Manager check box is available for the job definition.
 - If you do not select this check box, then the job cannot be run from Enterprise Manager.
 - If you select this check box, then you can define parameters for this job definition only in Enterprise Manager. Save the rest of your work on the job definition, and then go to Enterprise Manager if you need to define parameters.

Managing List of Values Sources: Highlights

A list of values source for Oracle Enterprise Scheduler job definitions determines where a list of values comes from and what the specific values are. These lists of values are used in parameters and user properties of job definitions. For example, you can use a source of country values for a Country job parameter.

Managing list of values sources is fully described in the Oracle Fusion Applications Administrator's Guide. As you read content from that guide, note that the guide describes managing Oracle Enterprise Scheduler, including list of values sources, from Oracle Enterprise Manager Fusion Applications Control. You can also access list of values sources by starting in the Setup and Maintenance Overview page and searching for Enterprise Scheduler job tasks.

Registering and Searching for List of Values Sources

- Create list of values sources to register them for use in job definitions.

See: Registering Sources for Lists of Values

- Search for list of values sources to edit or delete, or to make sure a particular source does not already exist before you create it.

See: Searching for List of Value Sources

Customization and Sandboxes

Customizing Pages Using Page Composer: Highlights

You can customize dashboards and some work areas, where available, for all or some users based on a selected customization layer, for example only for users in a specific country or with a specific job role. When you select to customize a page from the **Administration** menu in the global area, you invoke Page Composer, which enables the customization. The Administration menu is only available if you have appropriate roles.

From the same menu, you can also:

- Customize the global area.
- Manage customizations by accessing a list of components in the current page and details about the layers in which each component is customized.
- Access sandboxes to make customizations to a runtime use session before deploying your changes to the mainline.

Customizing pages using Page Composer, managing customizations, and using sandboxes are described in the Oracle Fusion Applications Extensibility Guide. For more information on customization layers, see assets with the Customization Layer type in Oracle Enterprise Repository for Oracle Fusion Applications (<http://fusionappsoer.oracle.com>).

Editing Pages

- Customizations that you can make to existing pages include adding content and changing layout.

See: Editing a Page in Page Composer

- You can also update display and other options for specific components within the page.

See: Editing Component Properties in Page Composer

Global Page Template

- Customize the global area at the site layer.

See: Editing the UI Shell Template Used by All Pages

Managing Customizations

- Use the Manage Customizations dialog box to analyze and diagnose customization metadata, and to perform customization related tasks that cannot be done in the user interface, for example to revert customizations to a previous version. You can also do direct customization by manipulating the metadata and uploading it back.

See: Viewing and Diagnosing Runtime Customizations

Sandboxes

- Create or select an appropriate sandbox, and set it as active to capture your customizations using Page Composer. When you are ready, you publish the sandbox to make your changes available to users.

See: Using the Sandbox Manager

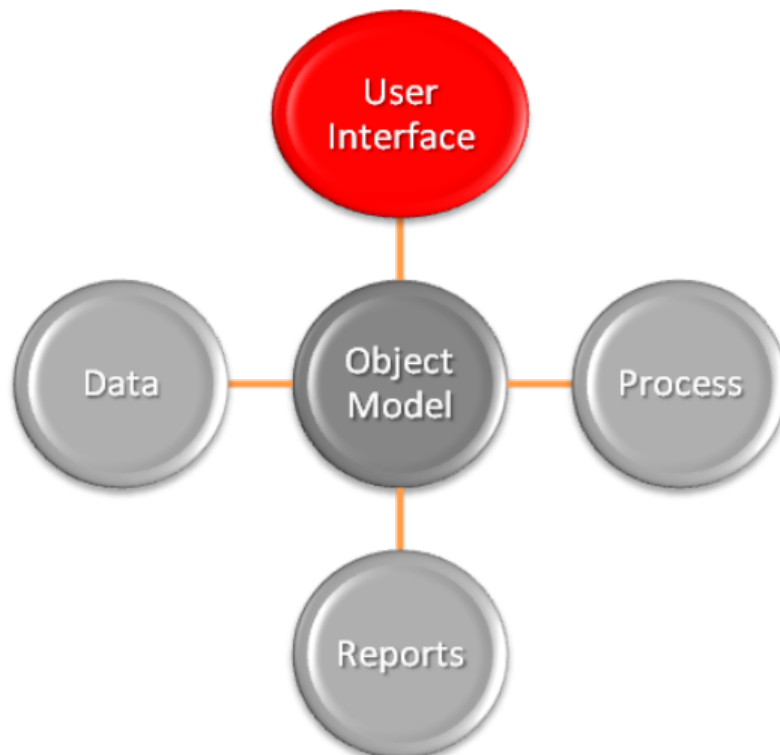
Customizing Oracle Fusion CRM Pages Using Page Composer: Explained

Page Composer is a page editor that lets you easily edit the user interface composition at run time. In Oracle Fusion CRM, Page Composer is intended for simple user interface editing functions, such as showing and hiding regions, fields, and tables, changing the order of regions, or changing a dashboard page layout. You can also use it for adding or removing predefined content from the Resource Library. All changes are done and stored in the UI layer. Oracle Fusion CRM transactional pages and dashboards are enabled for runtime customization using Page Composer. These pages and regions are delivered already enabled for page editing. Administration and setup pages are not Page Composer-enabled.

Note

Page Composer does not support the customization of custom object pages and custom fields. You can customize extended pages and fields only with the Application Composer.

The following figure shows how Page Composer affects only the UI layer.

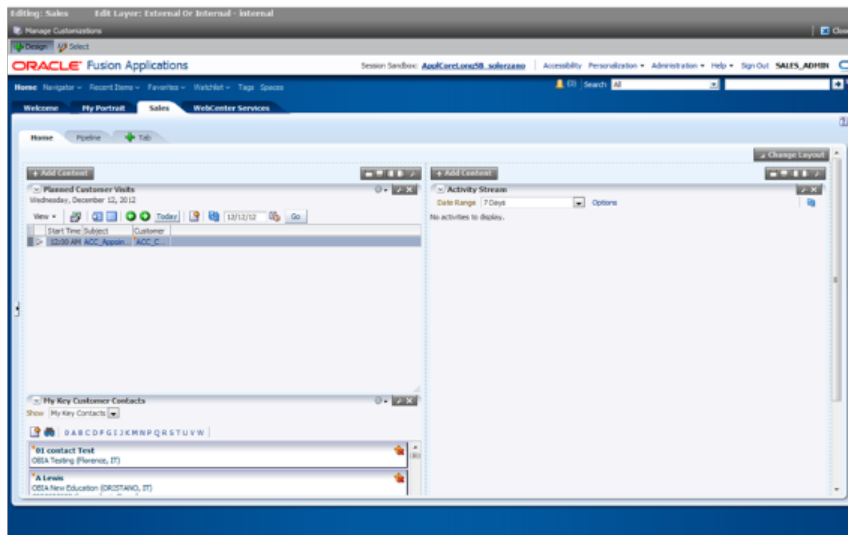


Page Composer Customization Modes

Page Composer supports two editing modes: Design View and Source View. In Design View mode, editing is done by direct manipulation of the target components and their properties. In the Source View mode, editing can be done from the programmatic code of the target component. In Oracle Fusion CRM, Page Composer is launched in Design View mode by default. Customizations can be done only in Design View mode. The only exception is the Partner Relationship Management's Partner Portal UI Shell customization, which is done in Source View mode.

The Page Composer Design View mode has two selection submodes. Design mode is launched by default when opening a page with Page Composer. You can also select this mode by clicking the Design tab on the top left corner of a page in Page Composer mode. Direct Selection mode is activated when you click the Select tab that is next to the Design tab.

The following figure shows a Page Composer-enabled dashboard in Design View mode and Design Selection mode.



The Direct Selection mode is mainly used to select and edit UI components, such as form fields and table columns. In Direct Selection mode, the enabled UI components become apparent when you move the cursor over the component. Enabled UI components get focus on hovering and are highlighted by a light blue border for editing. When you click the highlighted component, the border changes to dark blue and an Edit popup window appears.

The popup window allows you to select whether to edit the selected component or the parent component. Examples of parent components include tables that hold columns and forms that hold fields.

Note

In Oracle Fusion CRM, the Direct Selection mode is available when you customize pages through the Administration link, but not when you personalize them from the Personalization link.

Accessing Page Composer

You access Page Composer from the Administration global menu. Access to the Administration menu is controlled through the Administration Link View Duty. The following table lists examples of duty and job roles that must be assigned to you so that you can access the Page Composer editor.

Application	Duty Role	Job Role
CDH	Master Data Management Application Administrator	Master Data Management Application Administrator
Sales Catalog	Sales Catalog Administrator Duty	Sales Catalog Administrator
Sales	Sales Administrator Duty	Sales Administrator
PRM	Channel Administrator Duty	Channel Administrator
PRM	Channel Partner Portal Administrator Duty	Channel Partner Portal Administrator

Contact your security administrator for additional details on these security privileges.

Sandbox support is also available through Page Composer. The sandbox provides temporary storage for your customization changes until you are ready to commit them to the back end. Before working with sandboxes, review the guidelines and recommendations on using sandboxes, available in related help topics.

To launch Page Composer:

1. Navigate to the Page Composer-enabled page.
2. Before making changes using Page Composer, confirm that you are working in the desired sandbox. If not, create a sandbox before making your changes.
3. Select the Administration global menu.
4. On the Customize <page name> Pages popup dialog, select the MDS layer that you want to customize.
5. Click OK.
6. Click Customize <page name>.
7. Select **Customize Work Areas** for dashboard customization.

Page Composer is now launched.
8. Customize the page.
9. When you are done, click the **Close** button in the header to leave the Page Composer editor.

Available Customization Options

In Oracle Fusion CRM, you can use the following customization options:

- Transactional, landing, and dashboard page customizations. Only administrators can do these customizations.

- Partner Portal UI shell and branding customizations. Only Partner Portal administrators can do these customizations.
- User-level page personalization for limited personalization tasks, such as hiding or adding content. Personalization changes are only visible to the user who is doing them.

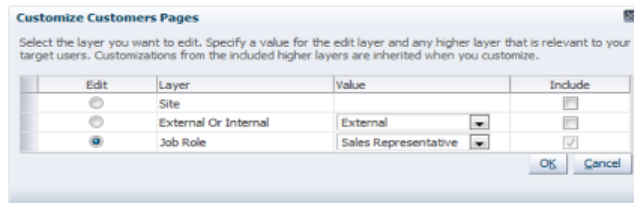
Additionally, you can do Page Composer customizations in any of these MDS customization layers:

- Site: Your customizations are visible to all site users.
- Job Role: Your customizations are visible to users who have the selected job role.
- External or Internal: Depending on your selection, your customizations are visible to either external or internal users. External users could be your partners or anonymous users. Internal users could be your employees.

When Page Composer is launched, the Customize <page name> Pages dialog opens, where you can select the MDS layer to customize. The layer that is selected in the Edit column is the layer that you want to edit. The layers that are selected in the Include column inherit any changes you make to the layer you edit.

By default, changes made at higher levels (Site) are propagated to lower level layers (External or Internal and Job Role), unless you uncheck these layers in the Include column.

The following figure shows selecting the MDS layer for customization.



Note

The Fusion Welcome dashboard does not support job role or External or Internal layer changes. All customizations to this dashboard are applied at the site level.

Available Customization Tasks

Design and Direct Selection modes support different sets of customization tasks (although in some instances a task can be performed in either mode). The Design mode is intended for overall UI composition customizations, such as adding content, changing the dashboard layout, and changing region properties. The Direct Selection mode is intended for customizations at the component rather than the region level, mainly for fields, forms, tables, and tree nodes customizations.

Tip

It might be necessary to toggle between Direct Selection and Design mode to navigate between pages when performing Direct Selection customizations, since you can only navigate in Design mode.

All Oracle Fusion CRM dashboard pages support the design mode of the Page Composer-enabled regions. External facing landing and transactional pages that are expected to need customizations also support the design mode for Page Composer-enabled regions:

- Partner Snapshot
- Edit Partner Profile
- Edit Partner Public Profile
- Edit Personal Profile
- Partner Landing
- Partner Registration Landing
- Partner Registration: Partner Information
- Partner Registration: Review and Accept Terms
- Partner Registration Information

In general, the following customizations are meant to be done using Direct Selection mode:

- Fields (input text fields, output fields, list of values fields):
 - Show or hide component
 - Make read-only
 - Make required
 - Edit label
- UI Containers (forms, tables and trees):
 - Show or hide child components (for example, fields in the form)
 - Reorder child components (for example, reorder fields in a form)

Note

UI components can be protected from updates to preserve the product business logic (for example, a field is read-only or not based on a security privilege already defined in the application). If protected from updates, the UI components are grayed out and cannot be customized in Page Composer.

The following table lists the customization tasks available through Page Composer and the modes in which they are supported.

Customization Task	Design View - Standard Mode	Design View - Direct Selection Mode
Change local area layout. For example, change a two-column layout to three-column layout (eight layouts are available).	Yes	No

Add, rename, remove dashboard tabs (except for Home tab).	Yes	No
Expand, collapse dashboard Regional pane (or move splitter location).	Yes	No
Add, remove panel boxes to or from the dashboard local area.	Yes	No
Add predefined content to dashboard panel boxes (Business Intelligence reports, CRM portlets, and common components such as Calendar).	Yes	No
Edit dashboard panel box properties: show, hide box, reorder child regions, display options, and style.	Yes	No
Add, remove ADF Components to or from dashboards (for example, regions, hyperlinks, images, text boxes, movable boxes, and Web pages) and edit their properties.	Yes	No
Customize saved searches (create and edit).	Yes	No
Hide or show field.	No	Yes
Change field label.	No	Yes
Make field required or not.	No	Yes
Make field read-only or updateable.	No	Yes
Reorder fields in a form.	No	Yes
Reorder table columns.	Yes	Yes
Hide or show table columns.	Yes (end users can optionally display columns at run time)	Yes (end users cannot display columns at run time)
Set table column width with the mouse.	Yes	No
Set table column width and minimum width in percent or pixels.	No	Yes
Enable, disable column sorting.	No	Yes

Customizing the Oracle Fusion CRM Welcome Dashboard

Also known as the Oracle Fusion CRM Home page, the Welcome dashboard is the application suite's default starting page. It is composed of a collection of tabs that are visible based on the roles assigned to users. The Welcome dashboard provides a collection of high-level data summaries meant for quick monitoring and navigation to key business objects. You can customize this page.

To customize the Welcome dashboard:

1. Navigate to the dashboard.
2. Click the Administration global menu.

3. Select the Customize Workarea Pages... menu item.

The Welcome dashboard consists of a local area only. The supported customization tasks, which you do in Design mode, are:

- Change local area layout.
- Add or remove panel boxes.
- Add seeded content to dashboard panel boxes.
- Edit dashboard panel box properties: show or hide a box, reorder child regions, change display options and style.
- Add, edit, and remove ADF components to or from dashboards, such as regions, hyperlinks, images, text boxes, movable boxes, and Web pages.

Note

The Welcome dashboard does not support role-based or External or Internal interface customization. All customizations are site-wide.

Oracle Fusion Common CRM Objects: Explained

You can customize a variety of pages and regions throughout Oracle Fusion CRM using the Oracle Fusion CRM Application Composer. The Application Composer lets you create custom fields and objects for a specific CRM application, which you then add for display at run time. Objects can be standard or custom, and are attached to a specific application container, such as Oracle Fusion Marketing or Oracle Fusion Sales. Oracle Fusion Common CRM objects, however, are unique in that the customizations you make to these objects can be displayed in a variety of CRM applications, and in a variety of pages and regions, at run time. This topic provides an overview of each Common CRM object that is part of the Trading Community Foundation. (This topic does not address the common component objects which are also listed within the Common CRM application container: the interaction, note, and task objects.)

The Trading Community Foundation Common CRM standard objects that you can extend are:

- Trading Community Address
- Trading Community Customer Contact Profile
- Trading Community Group Profile
- Trading Community Organization Profile
- Trading Community Person Profile
- Trading Community Relationship
- Trading Community Resource Profile

Note

Some of these objects are also documented in the related topic, Customizing Oracle Fusion Customer Center Pages: Explained.

Trading Community Address

The Trading Community Address object is an intersection object that stores a reference to a physical location/address, as well as a reference to a trading community organization, person, or group. Any extensions to this object are available for display in customer, contact, partner, and resource address user interfaces within Oracle Fusion CRM applications.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Address object.

This table indicates, for the Trading Community Address object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Address Creation Region	<ul style="list-style-type: none">• Edit Customer - Address Create region in Customer Center• Edit Contact - Address Create region in Customer Center• Edit Partner Profile - Address Create region in Partner Center• Edit Partner - Partner Members - Address Create region within Partner Center
Edit Address Summary Table	<ul style="list-style-type: none">• Edit Customer - Addresses region in Customer Center• Edit Contact - Addresses region in Customer Center• Edit Partner Profile - Address region in Partner Center• Edit Partner - Partner members - Address region within Partner Center
Edit Person Quick Creation Region	<ul style="list-style-type: none">• Consumer Create region within Customer Center• Contact Create region within Customer Center• Person Create region within Customer Hub
Edit Organization Quick Creation Region	<ul style="list-style-type: none">• Customer Create region within Customer Center• Organization Create region within Customer Hub

Note

Oracle Fusion CRM has provided multiple configuration capabilities for the Address object. Accordingly, do not use the Application Composer for these types of customizations:

- To define country-specific address formats, different labels for address elements, set address elements as required, and so on, use the Manage Address Formats task in the Setup and Maintenance work area.
 - To enable the list of values and validation for each country, use the Manage Geographies task in the Setup and Maintenance work area.
-

Trading Community Customer Contact Profile

The Trading Community Customer Contact Profile object stores all intrinsic information for all customer contact persons, in the context of the contact's relationship with a customer. For extensions specific to a customer contact relationship, use this object. By contrast, a standalone contact in Oracle Fusion CRM is a contact not tied to any customer and cannot be extended using this object. Instead, for extensions that apply to all person parties, use the Trading Community Person Profile object.

Any extensions to this object are available for display in various pages within Oracle Fusion CRM applications. Generic person extensions are also visible from the Oracle Fusion Customer Hub application in the Person work area.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Customer Contact Profile object.

This table indicates, for the Trading Community Customer Contact Profile object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Person Quick Creation Region	<ul style="list-style-type: none">• Consumer Create region within Customer Center• Contact Create region within Customer Center
Edit Contact Summary Table	<ul style="list-style-type: none">• Edit Customer - Contacts table in Customer Center• Edit Partner - Members table in Partner Center
Edit Contact Read-Only Region	<ul style="list-style-type: none">• Edit Contact - Basic Information region in Customer Center
Edit Contact Details Region	<ul style="list-style-type: none">• Edit Customer: Contact Details within Customer Center• Edit Organization: Relationships within Customer Hub• Edit Partner - Partner Members - Person Details region within Partner Center

Trading Community Group Profile

The Trading Community Group Profile object stores all intrinsic information for all groups within the trading community of the deploying company. In Oracle Fusion CRM, groups could be customers of the deploying company. The Trading Community Group Profile object is a generic profile that is applicable to all types of groups in the system. The Sales Account object is applicable only to those groups that are sales accounts. The Group Profile object is a date effective entity, which means that every change or update that happens to any attribute of the object causes a new record to be created in the object with effective start and end dates. This maintains the profile history information for the object.

Note

Extend this object only when your customization is required for all groups within your trading community.

In Oracle Fusion CRM, groups are available as customers, so any extensions to this object are available for display in the Customer pages. Generic group extensions are also visible from the Oracle Fusion Customer Hub application in the Groups work area. Extensions are not available for display in Customer Center.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Group Profile object.

This table indicates, for the Trading Community Group Profile object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Summary Table	<ul style="list-style-type: none">Oracle Fusion Customer Hub application in the Groups work area
Edit Group Update Form	<ul style="list-style-type: none">Oracle Fusion Customer Hub application in the Groups work area
Edit Group Create Form	<ul style="list-style-type: none">Oracle Fusion Customer Hub application in the Groups work area

Trading Community Organization Profile

The Trading Community Organization Profile object stores all intrinsic information for all organizations within the trading community of the deploying company. Examples of organizations from a CRM perspective are customers (including sales prospects and external legal entities), competitors, and partners. To specifically extend customer or partner objects, use the Sales Account object (under the Customer Center application container) or the Partner object (under the Sales application container). (The Trading Community Organization Profile object is a generic profile that is applicable to all types of organizations in the system. The Sales Account object is applicable only to those organizations that are sales accounts.) Since Prospect and Legal Entity customers don't have

separate profiles, you can extend these customers only by using the Trading Community Organization Profile object. The Organization Profile object is a date effective entity, which means that every change or update that happens to any attribute of the object causes a new record to be created in the object with effective start and end dates. This maintains the profile history information for the object.

Note

Extend this object only when your customization is required for all organizations within your trading community.

Any extensions to this object are available for display in various pages within Oracle Fusion CRM applications. Generic organization extensions are also visible from the Oracle Fusion Customer Hub application in the Organization work area.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Organization Profile object.

This table indicates, for the Trading Community Organization Profile object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Organization Details Region	<ul style="list-style-type: none"> • Customer Details region within Customer Center • Organization Details region within Customer Hub • Competitor Summary region within Sales
Edit Organization Quick Creation Region	<ul style="list-style-type: none"> • Customer Create region within Customer Center • Organization Create region within Customer Hub
Edit Organization Read-Only Region	<ul style="list-style-type: none"> • Basic Information Region within Customer Center • Basic Organization Information Region within Customer Hub

Trading Community Person Profile

The Trading Community Person Profile object stores all intrinsic information for all persons within the trading community of the deploying company. Examples of persons from a CRM perspective are customers (also called consumers), contacts, internal employee resources, and partner members. To extend the customer contact relationship, use the Trading Community Customer Contact Profile object. To extend internal resources, use the Trading Community Resource Profile object. To extend person customers (also called consumers), use

the Sales Account object (under the Customer Center application container). (The Trading Community Person Profile object is a generic profile that is applicable to all types of persons in the system. The Sales Account object is applicable only to those persons who are sales accounts.) The Person Profile object is a date effective entity, which means that every change or update that happens to any attribute of the object causes a new record to be created in the object with effective start and end dates. This maintains the profile history information for the object.

Note

Extend this object only when your customization is required for all persons within your trading community.

Any extensions to this object are available for display in various pages within Oracle Fusion CRM applications. Generic person extensions are also visible from the Oracle Fusion Customer Hub application in the Person work area.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Person Profile object.

This table indicates, for the Trading Community Person Profile object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Person Read-Only Region	<ul style="list-style-type: none"> • Basic Information region within Customer Center (for Consumer or Person customer) • Contact Basic Information within Customer Center • Basic Person Information region within Customer Hub
Edit Person Details Region	<ul style="list-style-type: none"> • Customer Details region within Customer Center (for Consumer or Person customer) • Manage Contacts: Contact Details within Customer Center
Edit Person Quick Creation Region	<ul style="list-style-type: none"> • Consumer Create region within Customer Center • Contact Create region within Customer Center • Person Create region within Customer Hub

Trading Community Relationship

The Trading Community Relationship object stores all intrinsic relationship information for all kinds of relationships between parties. Use this object to add attributes specific to a relationship between any two parties.

Note

Though a customer contact is also a type of relationship, use the Trading Community Customer Contact Profile object for customer contact relationship-specific extensions. Use this Trading Community Relationship object to extend only non-contact types of relationships.

Relationship Profile pages are regions that are visible in the Customer Hub. Accordingly, any extensions to this object are available for display in various pages within Oracle Fusion CRM applications. Generic person extensions are also visible from the Oracle Fusion Customer Hub application in the Person work area.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Relationship object.

This table indicates, for the Trading Community Relationship object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Relationship Summary Table	<ul style="list-style-type: none"> Edit Organization: Relationships within Customer Hub Edit Person: Relationships within Customer Hub Edit Group: Relationships within Customer Hub Create Group: Group Members within Customer Hub
Edit Relationship Creation Region	<ul style="list-style-type: none"> Edit Organization: Relationships within Customer Hub Edit Person: Relationships within Customer Hub Edit Group: Relationships within Customer Hub Create Group Members within Customer Hub

Trading Community Resource Profile

The Trading Community Resource Profile object defines users of the Oracle Fusion CRM application. They are either internal employee and contractor resources, or partner member resources. Use this object when specific extensions are required for persons in the system that are resources.

To display your extensions, such as custom fields, first create your custom fields using the Application Composer. Next, use the Application Composer's configuration pages to add those custom fields to the pages and regions listed below. You access the configuration pages in the Application Composer from the Pages node under the Trading Community Resource object.

This table indicates, for the Trading Community Resource object, which Application Composer configuration page lets you populate which page or region with your application customizations.

Configuration Page in Application Composer	Related Page or Region
Edit Details Region	<ul style="list-style-type: none"> The Resource Profile details page is available in the Resource Directory work area, or from the Manage Resources task in the Setup and Maintenance work area.
Edit Search Region for the Workarea Summary Table	<ul style="list-style-type: none"> The search region is available in the Resource Directory work area, under Search Resources.

Customizing CRM Applications Using the Application Composer: Explained

The Oracle Fusion CRM Application Composer provides a series of task flows which let you customize and extend an Oracle Fusion CRM application according to the needs of your users. For example, you can create fields for an existing standard object, and expose those new fields on the object's work area. Or, create a brand new custom object and related fields, then create a work area where that object and its fields are exposed to users. The task flows available to you are dependent upon the CRM application that you are customizing.

Available Customization Task Flows

Different sets of customization task flows are available to you, depending on the CRM application that you are customizing. See Customizing Oracle Fusion CRM Applications Using Oracle Fusion CRM Application Composer (Doc ID 1516151.1) on My Oracle Support at <https://support.oracle.com>. This document provides a list of which task flows are available for use in these CRM applications:

- Oracle Fusion Common CRM
 - This includes Master Data Management (MDM) and Common Party User Interface (CPUI) objects.
- Oracle Fusion Customer Center
- Oracle Fusion Marketing
- Oracle Fusion Sales
- Oracle Fusion Sales Catalog

You can also refer to the product-specific implementation guides to learn more about how a particular application works with the Application Composer.

Sandboxes: Highlights

Use a sandbox to commit customizations to a runtime use session for validation before deploying changes to the mainline. Administrators create and manage

sandboxes. An active sandbox isolates changes from the mainline and other users.

Sandboxes can contain the following types of customization changes.

- Metadata, such as non-flexfield UI page customizations
- Data security
- Generated flexfields business components

Metadata changes are captured in a metadata sandbox. Data security changes are additionally captured in a data security enabled sandbox. Changes to a flexfield are captured in a flexfield that is deployed as a single flexfield sandbox. Once you are ready to make sandbox changes available in the mainline, you either publish the metadata or data security sandbox, or deploy the flexfield. Only metadata and data security sandboxes can be downloaded as a sandbox file for import to another Oracle Fusion Applications instance.

The following table lists the differences among the types of sandboxes.

Type of Changes	Type of Sandbox	Method for Making Changes Available in Mainline	Downloadable?
Metadata	Sandbox	Publish sandbox	Yes
Data security	Sandbox enabled for data security changes	Publish sandbox	Yes
Flexfield	Flexfield deployed as a flexfield-enabled sandbox	Deploy flexfield	No

Only one sandbox can be active at a time. Changes made while a sandbox is active are captured in that sandbox.

For more information on using the Sandbox Manager, and customizing and securing pages, business objects, data, and custom objects in a sandbox, see the Oracle Fusion Applications Extensibility Guide.

Managing a Page Customization Sandbox

You can make metadata (MDS) type changes in a sandbox, including menu customizations, changes to the personalization menu, implicit ADF customizations, or changes made with Oracle Composer or CRM Application Composer.

- If you are entitled to do so, manage sandboxes in the Sandbox Manager.
See: Using the Sandbox Manager
- Implement customizations on an existing page to change the metadata of a sandbox before deploying the changes to the mainline.
See: Customizing Existing Pages
- Using CRM Application Composer, customize business objects in a sandbox before deploying the changes to the mainline.
See: Customizing Objects

Managing a Data Security Sandbox

You can create a sandbox for data security testing, or designate an existing sandbox to become enabled for data security testing.

- If you are entitled to do so, manage data security-enabled sandboxes in the Sandbox Manager.

See: Using the Sandbox Manager

- If you customize business objects in CRM Application Composer, you may need to define data security policies to make them accessible to users.

See: Defining Security Policies for Business Objects

- If you create new business objects, you need to secure them.

See: Customizing Security for Custom Business Objects

Managing a Flexfield Sandbox

You create a flexfield-enabled sandbox by deploying one flexfield to a sandbox using the Manage Flexfield task flow. The flexfield sandbox gets its name from the flexfield you deploy. You cannot test two flexfields in the same sandbox. Once you deploy a flexfield as a sandbox, you must sign out and back in to view how the sandbox runtime reflects the flexfield changes, such as new segments. You can redeploy the same flexfield to the same sandbox repeatedly as you make incremental changes to the flexfield setup.

- Since a flexfield sandbox cannot be published, any page customizations or data security in the flexfield sandbox cannot reach the mainline when the flexfield is deployed to the mainline. If you have entitlement to do so, see Deploying a Flexfield to a Sandbox: Points to Consider.
- If you are entitled to do so, manage flexfield-enabled sandboxes in the Sandbox Manager.

See: Using the Sandbox Manager

FAQs for Define Extensions for Customer Data Management

What's the difference between Page Composer and Application Composer?

Page Composer is an Oracle Fusion tool you can use to modify Oracle Fusion user interface (UI) pages and components for all products designated for use with Page Composer. Page Composer uses two different modes of Design View. The first mode, Design View: Standard mode, is selected by default in all CRM pages when opening a page with Page Composer with the Design button selected. The second mode, Design View: Direct Selection mode, is activated when you click the Select tab for the UI page you want to customize. In CRM, Direct Selection mode is available when you customize pages, but not when you personalize a dashboard page. With the Design View: Direct Selection mode, you can select and edit UI elements such as form fields and table columns. In Direct Selection mode, selectable UI components become apparent when you move your cursor over the UI component. Selectable UI components are highlighted and can be edited.

This table describes how you can use each mode of Page Composer to customize dashboard pages and other select pages (such as the Partner Public Profile page, Partner Landing page, Partner Registration, Customer Snapshot, and Customer Overview - Analysis tab), and customize transactional pages (all other non-dashboard pages):

Use Cases	Design View - Standard mode	Design View - Direct Selection mode	Page Type
Add content (Business Intelligence reports, CRM portlets such as Calendar)	Yes	No	Dashboard and other select pages
Delete region	Yes	No	Dashboard and other select pages
Move region	Yes	No	Dashboard and other select pages
Change page layout (for example, change a two column layout to three column layout)	Yes	No	Dashboard and other select pages
Default region state (open or close)	Yes	No	Transactional pages (all non-dashboard pages)
Manage save queries (create and edit)	Yes	No	Transactional pages (all non-dashboard pages)
Hide or show field	No	Yes	Transactional pages (all non-dashboard pages)
Change field label	No	Yes	Transactional pages (all non-dashboard pages)
Make field required or not	No	Yes	Transactional pages (all non-dashboard pages)
Make field read-only or updateable	No	Yes	Transactional pages (all non-dashboard pages)
Reorder fields in a Form	No	Yes	Transactional pages (all non-dashboard pages)
Reorder table columns	Yes	Yes	Transactional pages (all non-dashboard pages)
Hide or show table columns	Yes	Yes	Transactional pages (all non-dashboard pages)
Set table column width with the mouse	Yes	No	Transactional pages (all non-dashboard pages)
Set table column width and min width in percent or pixels	No	Yes	Transactional pages (all non-dashboard pages)
Make column sortable or not	No	Yes	Transactional pages (all non-dashboard pages)

The Oracle Fusion CRM Application Composer also lets you make UI changes at run time. However, the types of UI changes that you can make using the Application Composer are quite different. Specifically, your primary focus when using the Application Composer is to make actual object model changes. For

example, you can create a new business object and related fields, and then create new application pages where that object and its fields are exposed to users. The ability to make these types of object model extensions is available only in Oracle Fusion CRM applications. Also, using the Application Composer, you cannot access the Resource Catalog to add new content to a page. With Application Composer, administrators can make customizations at the site level only.

This table describes some of the primary differences between Page Composer and the Application Composer:

Customization Task	Available in Page Composer (site, job role, external or internal level)?	Available in Application Composer (site level only)?
Make object model extensions and expose your customizations by creating or modifying work area pages	No	Yes
Reorder subtabs	No	Yes
Customize dashboard pages	Yes	No
Add content from the Resource Catalog	Yes	No
Simple field customizations (show, hide, make read only, make required)	Yes (WYSIWYG - what you see is what you get)	Yes (non-WYSIWYG)
Select the MDS layer where you want to author customizations, such as at the site layer or job role layer	Yes	No
View results of customizations immediately	Yes, in the Page Composer design interface	Yes, in the CRM application that you are customizing

Importing and Exporting Setup Data

Configuration Packages: Explained

Almost all Oracle Fusion application implementations require moving functional setup data from one instance into another at various points in the lifecycle of the applications. For example, one of the typical cases in any enterprise application implementation is to first implement in a development or test application instance and then deploy to a production application instance after thorough testing. You can move functional setup configurations of applications from one application instance into another by exporting and importing Configuration packages from the Manage Configuration Packages page.

A Configuration Package contains the setup import and export definition. The setup import and export definition is the list of setup tasks and their associated business objects that identifies the setup data for export as well as the data itself. When you create a configuration package only the setup export and import definition exists. Once you export the configuration package appropriate setup data is added to the configuration package using the definition. Once a configuration package is exported, the setup export and import definition is locked and cannot be changed.

You generate the setup export and import definition by selecting an implementation project and creating a configuration package. The tasks and their associated business objects in the selected implementation project define the setup export and import definition for the configuration package. In addition, the sequence of the tasks in the implementation project determine the export and import sequence.

Exporting and Importing Setup Data: Explained

A configuration package is required to export setup data. You can export a configuration package once you create it, or at any time in the future. During export, appropriate setup data will be identified based on the setup export definition and added to the configuration package. The setup data in the configuration package is a snapshot of the data in the source application instance at the time of export. After the export completes, you can download the configuration package as a zipped archive of multiple XML files, move it to the target application instance, and upload and import it.

Export

You can export a configuration package multiple times by creating multiple versions. While the export definition remains the same in each version, the setup

data can be different if you modified the data in the time period between the different runs of the export process. Since each version of the configuration package has a snapshot of the data in the source instance, you can compare and analyze various versions of the configuration package to see how the setup data changed.

Import

In the target application instance, the setup import process will insert all new data from the source configuration package that does not already exist and update any existing data with changes from the source. Setup data that exists in the target instance but not in source will remain unchanged.

Export and Import Reports

You can review the results of the export and import processes using reports. The results appear ordered by business objects and include information on any errors encountered during the export or import process. If a setup export or import process paused due to errors encountered or for a manual task to be performed outside of the application, then you can resume the paused process.

These reports show what setup data was exported or imported and by which specific process. You can change the reports to validate the setup data as well as to compare or analyze it. A report is generated for each business object. These reports show the same information as the export and import results seen directly in the application.

Process status details are available as text files showing the status of an export or import process including the errors encountered during the process.

Moving Common Reference Objects

Moving Common Reference Objects: Overview

The common reference objects in Oracle Fusion Middleware Extensions for Applications are used by several setup tasks in the Setup and Maintenance work area. The common reference objects become a part of the configuration package that is created for an implementation project. While moving the application content, for example, from the test phase to the production phase of an implementation, you must pay special attention to the nuances of these common reference objects.

Parameters

The common reference objects are represented as business objects. A single object can be referenced in multiple setup tasks with different parameters. In the configuration package that is created for the implementation project, parameters passed to a setup task are also passed to the business objects being moved. As a result, the scope of the setup tasks is maintained intact during the movement.

Dependencies

Common reference objects may have internal references or dependencies among other common reference objects. Therefore, it is necessary that all the dependencies are noted before the movement of objects so that there are no broken references among the objects.

Business Objects for Moving Common Reference Objects: Points to Consider

Common reference objects in Oracle Fusion Functional Setup Manager are represented by business objects. These business objects are the agents that contain the application content and carry them across whenever the application setup is moved from one environment to another, for example, test environment to production environment.

Choice of Parameters

The following table lists the business objects, the corresponding movement details, and the effect of the setup task parameter on the scope of the movement.

Note

- Only the translation in the current user language is moved.
- 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.

Business Object Name	Moved Functional Item	Effect on the Scope of Movement
Application Message	Messages and associated tokens	No parameters: all messages are moved. moduleType/moduleKey: only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved. messageName/applicationId: only the specified message is moved.
Application Taxonomy	Application taxonomy modules and components	No parameters: all taxonomy modules and components are moved.
Application Attachment Entity	Attachment entities	No parameters: all attachment entities are moved. moduleType/moduleKey: only attachment entities belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
Application Attachment Category	Attachment categories and category-to-entity mappings	No parameters: all attachment categories and category-to-entity mappings are moved. moduleType/moduleKey: only attachment categories belonging to the specified module and its descendant modules in the taxonomy hierarchy along with the respective category-to-entity mappings are moved.

Application Document Sequence Category	Document sequence categories	<p>No parameters: all categories are moved.</p> <p><code>moduleType/moduleKey</code>: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>code/applicationId</code>: only the specified document sequence category code is moved.</p>
Application Document Sequence	Document sequences and their assignments	<p>No parameters: all sequences are moved.</p> <p><code>moduleType/moduleKey</code>: only document sequences belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved</p> <p><code>name</code>: only the specified document sequence is moved.</p>
Application Descriptive Flexfield	Descriptive flexfield registration data and setup data	<p>No parameters: all descriptive flexfields are moved.</p> <p><code>moduleType/moduleKey</code>: only descriptive flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>descriptiveFlexfieldCode/applicationId</code>: only the specified descriptive flexfield is moved.</p> <hr/> <p>Note</p> <p>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.</p> <hr/> <p>Note</p> <p>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</p> <hr/>

Application Extensible Flexfield	Extensible flexfield registration data and setup data, including categories	<p>No parameters: all extensible flexfields are moved</p> <p><code>moduleType/moduleKey</code>: only extensible flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>extensibleFlexfieldCode/applicationId</code>: only the specified extensible flexfield is moved.</p> <hr/> <p>Note</p> <p>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.</p> <hr/> <p>Note</p> <p>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</p>
Application Key Flexfield	Key flexfield registration data and setup data	<p>No parameters: all key flexfields are moved.</p> <p><code>moduleType/moduleKey</code>: only key flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>keyFlexfieldCode/applicationId</code>: only the specified key flexfield is moved.</p> <hr/> <p>Note</p> <p>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.</p> <hr/> <p>Note</p> <p>Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.</p>

Application Flexfield Value Set	Value set setup data	<p>No parameters: all value sets are moved.</p> <p><code>moduleType/moduleKey</code>: only value sets belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>valueSetCode</code>: only the specified value set is moved.</p> <hr/> <p>Note</p> <p>Importing a value set's metadata can change the deployment status of flexfields that use the value set, and therefore the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.</p>
Application Reference Currency	Currency data	No parameters: all currencies are moved.
Application Reference ISO Language	ISO language data	No parameters: all ISO languages are moved.
Application Reference Industry	Industry data including industries in territories data	No parameters: all industries are moved.
Application Reference Language	Language data	No parameters: all languages are moved.
Application Reference Natural Language	Natural language data	No parameters: all natural languages are moved.
Application Reference Territory	Territory data	No parameters: all territories are moved.
Application Reference Time zone	Time zone data	No parameters: all time zones are moved.
Application Standard Lookup	Standard lookup types and their lookup codes	<p>No parameters: all standard lookups are moved.</p> <p><code>moduleType/moduleKey</code>: only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>lookupType</code>: only the specified common lookup is moved.</p>

Application Common Lookup	Common lookup types and their lookup codes	<p>No parameters: all common lookups are moved.</p> <p><code>moduleType/moduleKey</code> - only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>lookupType</code>: only the specified common lookup is moved.</p>
Application Set-Enabled Lookup	Set-enabled lookup types and their lookup codes	<p>No parameters: all set-enabled lookups are moved.</p> <p><code>moduleType/moduleKey</code>: only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>lookupType</code>: only the specified set-enabled lookup is moved.</p>
Application Profile Category	Profile categories	<p>No parameters: all profile categories are moved.</p> <p><code>moduleType/moduleKey</code>: only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>name/applicationId</code>: only the specified category is moved.</p>
Application Profile Option	Profile options and their values	<p>No parameters: all profile options and their values are moved.</p> <p><code>moduleType/moduleKey</code>: only profile options and their values belonging to the specified module are moved.</p> <p><code>profileOptionName</code>: only the specified profile option and its values are moved.</p>

Application Profile Value	Profile options and their values	<p>No parameters: all profiles and their values are moved.</p> <p><code>moduleType/moduleKey</code>: only profiles and their values belonging to the specified module are moved.</p> <p><code>categoryName/</code> <code>categoryApplicationId</code>: only profiles and their values belonging to the specified category are moved.</p> <p><code>profileOptionName</code>: only the specified profile and its values are moved.</p>
Application Reference Data Set	Reference data sets	No parameters: all sets are moved.
Application Reference Data Set Assignment	Reference data set assignments	<p><code>determinantType</code>: only assignments for the specified determinant type are moved.</p> <p><code>determinantType/</code> <code>referenceGroupName</code>: only assignments for the specified determinant type and reference group are moved.</p>
Application Tree Structure	Tree structures and any labels assigned to the tree structure	<p>No parameters: all tree structures (and their labels) are moved.</p> <p><code>moduleType/moduleKey</code>: only tree structures (and their labels) belonging to the specified module are moved.</p> <p><code>treeStructureCode</code>: only the specified tree structure (with its labels) is moved.</p>
Application Tree	Tree codes and versions	<p>No parameters: all trees are moved.</p> <p><code>moduleType/moduleKey</code>: only trees belonging to the specified module are moved.</p> <p><code>treeStructureCode</code>: only trees belonging to the specified tree structure are moved.</p> <p><code>TreeStructureCode/TreeCode</code>: only trees belonging to the specified tree structure and tree code are moved.</p>

Application Tree Label	Tree structures and any labels assigned to the tree structure	<p>No parameters: all tree structures (and their labels) are moved.</p> <p><code>moduleType/moduleKey</code>: only tree structures (and their labels) belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>treeStructureCode</code>: only the specified tree structure (with its labels) is moved.</p>
Application Data Security Policy	Database resources, actions, conditions, and data security policies	<p>No parameters: all database resources/actions/conditions/policies are moved.</p> <p><code>moduleType/moduleKey</code>: only database resources/actions/conditions/policies belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.</p> <p><code>objName</code>: only the specified database resource along with its actions/conditions/policies is moved.</p> <hr/> <p>Note</p> <ul style="list-style-type: none"> • If the policies being moved contain reference to newly created roles, move the roles before moving the policies. • If the source and target systems use different LDAPs, manually perform the GUID reconciliation after moving the data security policies.
Application Activity Stream Configuration	Activity stream options	No parameters: all activity stream options are moved.

Moving Related Common Reference Objects: Points to Consider

Certain common reference objects may use other common reference objects creating dependencies among the objects. During the movement of common reference objects, these dependencies or references need to be taken care of.

Dependencies

The dependencies among the common reference objects may be caused by any of the following conditions.

- Flexfield segments use value sets

- Value sets may make use of standard, common, or set-enabled lookups
- Key flexfields may have an associated tree structure and key flexfield segments may have an associated tree code
- Tree codes and versions may be defined over values of a value set
- Data security policies may be defined for value sets that have been enabled for data security

You may choose to move one, some, or all of the business objects by including the ones you want to move in your configuration package. For example, you may choose to move only value sets and not lookups, or you may choose to move both value sets and their lookups as part of the same package. Whatever be the combination, it is recommended that during the movement of objects, you follow an order that maintains the dependencies among the objects.

While moving the business objects, adhere to the guidelines and exactly follow the order as listed below.

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.

Note

In scenarios where there may be dependencies on other objects, you must move the dependencies before moving the referencing object. For example, if data security policies being moved have dependencies on newly created security roles, you must move the security roles before moving the security policies.

Using Seed Data Framework to Move Common Reference Objects: Points to Consider

To move the common reference objects, you can use the Seed Data Framework (SDF). You can also use the command line interface of SDF to move the object

setup data. For more information about seed data loaders including common reference object loaders, see Oracle Fusion Applications Developer's Guide.

Movement Dependencies

The seed data interface moves only the setup metadata. For example, if you use SDF to import flexfield metadata, the flexfield setup metadata is imported into your database. However, you must invoke the flexfield deployment process separately after seed data import to regenerate the runtime flexfield artifacts in the target environment. Similarly, if you use SDF to import data security metadata, you must first move any new referenced roles and then manually run the GUID reconciliation where required.

To ensure that the reference data is not lost during the movement, certain guidelines are prescribed. It is recommended that you perform the movement of object data exactly in the order given below.

Note

Only the translation in the current user language is moved.

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, attachments, reference data sets, document sequences, messages, and data security.
2. Move created currencies before moving any objects that reference them, such as territories.
3. Move created territories before moving any objects that reference them, such as languages and natural languages.
4. Move created ISO languages before moving any objects that reference them, such as languages, natural languages, and industries.
5. Move created tree structures before moving any objects that reference them, such as trees or tree labels.
6. Move created profile options before moving any objects that reference them, such as profile categories or profile values.
7. Move created attachment entities before moving any objects that reference them, such as attachment categories that reference them.
8. Move created reference data sets before moving any objects that reference them, such as reference data set assignments and set-enabled lookups.
9. Move created document sequence categories before moving any objects that reference them, such as document sequences.
10. Move created tree labels before moving any objects that reference them, such as trees.
11. Move created data security objects and policies before moving any objects that reference them, such as value sets.
12. Move created value sets before moving any objects that reference them, such as flexfields.
13. Move created trees before moving any objects that reference them, such as key flexfields.

Glossary

abstract role

A description of a person's function in the enterprise that is unrelated to the person's job (position), such as employee, contingent worker, or line manager. A type of enterprise role.

accounting flexfield

The chart of accounts that determines the structure, such as the number and order of individual segments, as well as the corresponding values per segment.

action

The kind of access named in a security policy, such as view or edit.

address style format

Specifies the layout of an address, such as how many address lines it contains, and whether a city name is mandatory.

ADF

Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

application identity

Predefined application level user with elevated privileges. An application identity authorizes jobs and transactions for which other users are not authorized, such as a payroll run authorized to access a taxpayer ID while the user who initiated the job is not authorized to access such personally identifiable information.

application module

An application module is the transactional component that UI clients use to work with application data. It defines an updatable data model and top-level procedures and functions (called service methods) for a logical unit of work related to an end-user task.

application module class

Includes service methods or extended code required by the application module configuration.

application role

A role specific to applications and stored in the policy store.

Applications Core

Abbreviation for Oracle Fusion Middleware Extensions for Applications. The technical product code is FND.

assignment

A set of information, including job, position, pay, compensation, managers, working hours, and work location, that defines a worker's or nonworker's role in a legal employer.

balancing segment

A chart of accounts segment used to automatically balance all journal entries for each value of this segment.

beneficiary

A person or organization designated to receive benefits from a compensation plan on the death of the plan participant.

BPEL

Business Process Execution Language; a standard language for defining how to send XML messages to remote services, manipulate XML data structures, receive XML messages asynchronously from remote services, manage events and exceptions, define parallel sequences of execution, and undo parts of processes when exceptions occur.

business function

A business process, or an activity that can be performed by people working within a business unit and describes how a business unit is used.

business object

A resource in an enterprise database, such as an invoice or purchase order.

business unit

A unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy.

calendar event

A period that signifies an event, such as a public holiday or a training course, that impacts worker availability.

chart of accounts

The account structure your organization uses to record transactions and maintain account balances.

clause adoption

Reusing a clause from the global business unit in local business units either by adopting the clause without change or by localizing it.

clause localization

A type of clause adoption where the adopted clause is edited to suit the local business unit needs.

condition

An XML filter or SQL predicate WHERE clause in a data security policy that specifies what portions of a database resource are secured.

context

A grouping of flexfield segments to store related information.

context segment

The flexfield segment used to store the context value. Each context value can have a different set of context-sensitive segments.

context-sensitive segment

A flexfield segment that may or may not appear depending upon a context such as other information that has been captured. Context-sensitive segments are custom attributes that apply to certain entity rows based on the value of the context segment.

contingent worker

A self-employed or agency-supplied worker. Contingent worker work relationships with legal employers are typically of a specified duration. Any person who has a contingent worker work relationship with a legal employer is a contingent worker.

contract deviations

Differences between the contract terms in a contract and those in the contract terms template applied to that contract and any deviations from company policies as determined by Contract Expert feature rules.

Contract Expert

A feature of the application that permits you to create business rules in the Contract Terms Library to enforce corporate policies and standards for contracts.

Contract Terms Library

A repository of standard clauses, contract terms templates, and business rules built using Contract Expert.

corporate rate type

Rate you define to standardize rates used in conversion of one currency to another over a period of time. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.

cost center

A unit of activity or group of employees used to assign costs for accounting purposes.

cost organization

A grouping of inventory organizations that indicates legal and financial ownership of inventory, and which establishes common costing and accounting policies.

country holding company

A legal entity that acts on behalf of several divisions within an enterprise, and is the legal employer in a country.

data dimension

A stripe of data accessed by a data role, such as the data controlled by a business unit.

data instance set

The set of human capital management (HCM) data, such as one or more persons, organizations, or payrolls, identified by an HCM security profile.

data role

A role for a defined set of data describing the job a user does within that defined set of data. A data role inherits job or abstract roles and grants entitlement to access data within a specific dimension of data based on data security policies. A type of enterprise role.

data role template

A template used to generate data roles by specifying which base roles to combine with which dimension values for a set of data security policies.

data security

The control of access to data. Data security controls what action a user can taken against which data.

data security policy

A grant of entitlement to a role on an object or attribute group for a given condition.

database resource

An applications data object at the instance, instance set, or global level, which is secured by data security policies.

department

A division of a business enterprise dealing with a particular area of activity.

descriptive flexfield

Customizable expansion space, such as fields used to capture additional descriptive information or attributes about an entity, such as customer cases. Information collection and storage may be configured to vary based on conditions or context.

determinant

A value that determines which reference data set will be used in a specific business context.

determinant type

Designates the field within transactional columns that controls how data is shared across organizations such as business unit, asset book, cost organization or project unit. The type determines the reference data sets that would be used in a transaction.

determinant type

An additional and optional field within transactional columns (besides category and application) that is used to assign document sequences. The available determinant types are Business Unit, Ledger, Legal Entity, and Tax Registration.

determinant value

A value specific to the determinant type dimension of a document sequence. The determinant value is relevant in a document sequence assignment only if the document sequence has a determinant type. If Ledger is the determinant type for a document sequence, the determinant value is the specific ledger number whose documents are numbered by the document sequence.

division

A business-oriented subdivision within an enterprise. Each division is organized to deliver products and services or address different markets.

document sequence

A unique number that is automatically or manually assigned to a created and saved document.

document type

A categorization of person documents that provides a set of options to control what document information to retain, who can access the documents, whether the documents require approval, and whether the documents are subject to expiry. A document type exists for a combination of document category and subcategory.

duty role

A group of function and data privileges representing one duty of a job. Duty roles are specific to applications, stored in the policy store, and shared within an Oracle Fusion Applications instance.

employment terms

A set of information about a nonworker's or employee's job, position, pay, compensation, working hours, and work location that all assignments associated with the employment terms inherit.

enterprise

An organization with one or more legal entities under common control.

enterprise role

Abstract, job, and data roles are shared across the enterprise. An enterprise role is an LDAP group. An enterprise role is propagated and synchronized across Oracle Fusion Middleware, where it is considered to be an external role or role not specifically defined within applications.

entitlement

Grants of access to functions and data. Oracle Fusion Middleware term for privilege.

extensible flexfield

Customizable expansion space, as with descriptive flexfields, but able to capture multiple sets of information within a context and multiple contexts grouped to appear in a named region of a user interface page. Some extensible flexfields allow grouping contexts into categories.

feature choice

A selection you make when configuring offerings that modifies a setup task list, or a setup page, or both.

fixed rate type

Rate you set between two currencies that remains constant. For example, a rate set between the euro currency and each Economic and Monetary Union (EMU) currency during the conversion to the euro currency.

flexfield

Grouping of extensible data fields called segments, where each segment is an attribute added to an entity for capturing additional information.

flexfield segment

An extensible data field that represents an attribute on an entity and captures a single atomic value corresponding to a predefined, single extension column in

the Oracle Fusion Applications database. A segment appears globally or based on a context of other captured information.

function security

The control of access to a page or a specific widget or functionality within a page. Function security controls what a user can do.

gallery

A searchable collection of portraits that combines the functions of the person directory with corporate social networking and self-service applications for both workers and managers.

global area

The region across the top of the user interface. It provides access to features and tools that are relevant to any page you are on.

grade

A component of the employment model that defines the level of compensation for a worker.

HCM data role

A job role, such as benefits administrator, associated with specified instances of Oracle Fusion Human Capital Management (HCM) data, such as one or more positions or all persons in a department.

HCM securing object

An HCM object that secures access to both its own data and data in other, related objects. For example, access to a specified set of person records can allow access to data secured by person records, such as goal plans and evaluations.

hierarchy version

Hierarchy version refers to a specific version of a hierarchy. You can create any number of versions, but only one hierarchy version can be active on a specific date. In tree terminology, hierarchy versions are also called tree versions.

identifier type

An identifier type is a specific category of additional identifiers, numeric or otherwise, which can identify trading community members. Examples of identifier types could be bank account numbers, passport numbers, and so on.

identity

A person representing a worker, supplier, or customer.

interface table

A database table used for transferring data between applications or from an external application or data file.

inventory organization

An organization that tracks inventory transactions and balances, and can manufacture or distribute products.

item master

A collection of data that describes items and their attributes recorded in a database file.

Items

Entries within the Product master database. For example, items for a manufacturing company can include nuts, bolts, and screws.

job

A generic role that is independent of any single department or location. For example, the jobs Manager and Consultant can occur in many departments.

job role

A role for a specific job consisting of duties, such as an accounts payable manager or application implementation consultant. A type of enterprise role.

key flexfield

Configurable key consisting of multiple parts or segments, each of which may be meaningful individually or in combination with the others. Key flexfields are commonly implemented to represent part numbers and account numbers.

key flexfield segment instance

A single occurrence of a key flexfield segment in a key flexfield structure instance.

key flexfield structure

The arrangement of segments in a key flexfield. In some cases, multiple structures can be defined for a single key flexfield.

key flexfield structure instance

A single occurrence of a key flexfield structure that shares the same order of segments as every other instance of the key flexfield structure, but uses different value sets to validate the segments.

legal employer

A legal entity that employs people.

legal entity

An entity is identified and given rights and responsibilities under commercial law, through the registration with the country's appropriate authority.

legal reporting unit

The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.

legislative data group

A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.

line of business

Set of one or more highly related products which service a particular customer transaction or business need. Refers to an internal corporate business unit.

lookup code

A value available for lookup within a lookup type such as the code BLUE within the lookup type COLORS.

lookup type

A set of lookup codes to be used together as a list of values on a field in the user interface.

mainline

A branch of data that serves as a single source of truth.

managed person

In Oracle Fusion Human Capital Management security, a person for whom the user can maintain some information. For example, line managers can maintain information about their direct and indirect reports, and workers can maintain information about themselves, their dependents, and their beneficiaries.

marketing list

A static selection of contacts for the purpose of communicating a marketing message by e-mail, direct mail or phone.

model profile

A collection of the work requirements and required skills and qualifications of a workforce structure, such as a job or position.

name style format

Specifies the layout of a name, such as first name, last name, and phonetic last name.

name type

A name type is a specific category of additional names that can identify trading community members. Examples of name types could be maiden names, aliases, doing-business-as names, and so on.

natural account

Categorizes account segment values by account type, asset, liability, expense, revenue, or equity, and sets posting, budgeting, and other options.

offering

A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.

organization hierarchy

A tree structure that determines the relationships between organizations, such as which organizations are subordinate to other organizations.

OWLCS

Abbreviation for Oracle WebLogic Communication Services. Offers the TPCC service to Fusion applications and sets up the calls via SIP integration with the telephony network.

party

A physical entity, such as a person, organization or group, that the deploying company has an interest in tracking.

payroll statutory unit

A legal entity registered to report payroll tax and social insurance. A legal employer can also be a payroll statutory unit, but a payroll statutory unit can represent multiple legal employers.

pending worker

A person who will be hired or start a contingent worker placement and for whom you create a person record that is effective before the hire or start date.

person number

A person ID that is unique in the enterprise, allocated automatically or manually, and valid throughout the enterprise for all of a person's work and person-to-person relationships.

person type

A subcategory of a system person type, which the enterprise can define. Person type is specified for a person at the employment-terms or assignment level.

personally identifiable information

Any piece of information that can potentially be used to uniquely identify, contact, or locate a single person. Within the context of an enterprise, some PII data can be considered public, such as a person's name and work phone number, while other PII data is confidential, such as national identifier or passport number.

PL/SQL

Abbreviation for procedural structured queried language.

portrait

A selection of information about a worker or nonworker, including contact details, social connections, and activities and interests, that can be viewed and edited. Both the amount and type of information and the available actions depend on the role of the portrait user.

position

A specific occurrence of one job, fixed within one department, also often one location. For example, the position Finance Manager is an instance of the job Manager in the Finance Department.

primary ledger

Main record-keeping ledger.

privilege

A grant or entitlement of access to functions and data. A privilege is a single, real world action on a single business object.

profile option

User preferences and system configuration options consisting of a name and a value, that can be set at hierarchical levels of an enterprise. Also called a profile or user option.

profile option level

A level at which profile option values are defined. Site, product, and user are predefined levels.

profile option level hierarchy

The ordering of profile option levels. The order of the levels in the hierarchy determines which levels take precedence.

profile option value

The value portion of a profile option's name and value. A profile option may have multiple values set at different levels, such as site or user.

project expenditure organization

An organization that can incur expenditures and hold financial plans for projects.

PSTN

Abbreviation for public switched telephone network which is the network of the world's public circuit-switched telephone networks.

public person

In Oracle Fusion Human Capital Management security, a person for whom some basic information is publicly available. For example, users typically access the contact details of public persons, such as phone numbers and locations, using the person gallery.

reference data

Data in application tables that is not transactional and not high-volume such as sales methods, transaction types, or payment terms, and can be shared and used across organizational boundaries.

reference data set

Contains reference data that can be shared across a number of business units or other determinant types. A set supports common administration of that reference data.

reference group

A logical grouping of tables that correspond to logical entities such as payment terms defined across multiple tables or views. Grouping establishes common partitioning requirements across the entities causing them to share the same set assignments.

resource

People designated as able to be assigned to work objects, for example, service agents, sales managers, or partner contacts. A sales manager and partner contact can be assigned to work on a lead or opportunity. A service agent can be assigned to a service request.

resource organization

An organization whose members are resources. Resource organizations are used to implement sales organizations, partner organizations, and so on.

resource role

Resource roles indicate the role a resource plays as an individual, or within a resource team.

resource team

A resource team is a temporary group of resources formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization.

role

Controls access to application functions and data.

role hierarchy

Structure of roles to reflect an organization's lines of authority and responsibility. In a role hierarchy, a parent role inherits all the entitlement of one or more child roles.

role mapping

A relationship between one or more job roles, abstract roles, and data roles and one or more conditions. Depending on role-mapping options, the role can be provisioned to or by users with at least one assignment that matches the conditions in the role mapping.

role provisioning

The automatic or manual allocation of an abstract role, a job role, or a data role to a user.

sandbox

A runtime session that commits changes out of reach of mainline users.

security profile

A set of criteria that identifies one or more human capital management (HCM) objects of a single type for the purposes of securing access to those objects. Security profiles can be defined for persons, organizations, positions, countries, LDGs, document types, payrolls, payroll flows, and workforce business processes.

security reference implementation

Predefined function and data security in Oracle Fusion Applications, including role based access control, and policies that protect functions, data, and segregation of duties. The reference implementation supports identity management, access provisioning, and security enforcement across the tools, data transformations, access methods, and the information life cycle of an enterprise.

segregation of duties

An internal control to prevent a single individual from performing two or more phases of a business transaction or operation that could result in fraud.

service provider model

A business unit that provides specific business functions for another business unit.

set

Reference data that is organized into groups appropriate to organizational entities, to enable reference data sharing.

set enabled

An entity, such as a lookup, customer, location, organization, or document attachment, that is allowed to participate in reference data sharing by drawing on the data of a reference data set.

SOA

Abbreviation for service-oriented architecture.

spot rate type

Rate you enter to perform conversion based on this rate as of a specific date. This rate applies to the immediate delivery of a currency.

SQL predicate

A type of condition using SQL to constrain the data secured by a data security policy.

survivorship

The process of creating the best version record intelligently, especially from multiple source systems, by specifying criteria for selecting record and attributes to be retained during merge or link, or update operations.

system person type

A fixed name that the application uses to identify a group of people.

territory

A legally distinct region that is used in the country field of an address.

tree

Information or data organized into a hierarchy with one or more root nodes connected to branches of nodes. A tree must have a structure where each node corresponds to data from one or more data sources.

tree structure

Characteristics applied to trees, such as what data to include or how the tree is versioned and accessed.

tree version

An instance of a tree. If a tree is associated with a reference data set, all versions belong to one set. Includes life cycle elements such as start and end date and a status indicator whether the tree is active or not.

user rate type

Rate you enter at journal entry time to convert foreign currency transactions to your functional currency.

value set

A set of valid values against which values entered by an end user are validated. The set may be tree structured (hierarchical).

work relationship

An association between a person and a legal employer, where the worker type determines whether the relationship is a nonworker, contingent worker, or employee work relationship.

worker type

A classification selected on a person's work relationship, which can be employee, contingent worker, pending worker, or nonworker.

workflow

An automated process in which tasks are passed from a user, a group of users, or the application to another for consideration or action. The tasks are routed in a logical sequence to achieve an end result.

workforce business process

A defined sequence of tasks, registered for the enterprise, that delivers a major business goal. The tasks may be allocated to multiple roles across multiple products.

XML

Abbreviation for eXtensible markup language.

XML filter

A type of condition using XML to constrain the data secured by a data security policy.