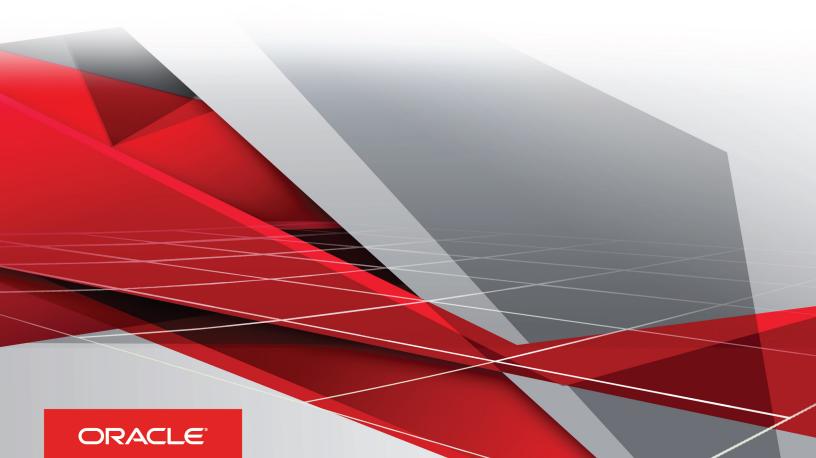
# Oracle

# Financials Cloud Implementing Common Features

Release 12

This guide also applies to on-premises implementations



Oracle® Financials Cloud Implementing Common Features

Part Number E73063-02

Copyright © 2011-2017, Oracle and/or its affiliates. All rights reserved.

Authors: Scott Dunn, Essan Ni Jirman, Barnali Roy, P. S. G. V. Sekhar, Robert MacIsaac

Contributors: Kristy Carlile, Rakesh Chaudhary, Suzanne Kinkead, Michael Laverty, Reshma Shaik, Srinivas Vellikad, Megan Wallace, 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, then 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 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 about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customers that have purchased support 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.

### **Contents**

Preface	i
Overview	1
Common Implementation: Overview	1
Understanding Implementation Structures	1
Managing an Implementation	4
Maintain Common Reference Objects	7
Maintain Common Reference Objects: Overview	7
Why can't I edit setup data?	7
Define Application Taxonomy	7
Define Reference Data Sharing	9
Define ISO Reference Data	13
Manage Audit Policies	16
Manage Oracle Social Network Objects	18
Manage Applications Core Common Reference Objects	22
Define Global Search	25
Define Currencies and Currency Rates	35
Manage Currencies	35
Manage Conversion Rate Types	36



4	Define Enterprise Structures	41
	Enterprise Structures: Overview	41
	Enterprise Structures Business Process Model: Explained	43
	Global Enterprise Configuration: Points to Consider	45
	Modeling Your Enterprise Management Structure in Oracle Fusion: Example	46
	Essbase Character and Word Limitations	50
	Define Initial Configuration with the Enterprise Structures Configurator	53
	Define Reference Data Sharing	80
	Define Enterprise: Manage Enterprise HCM Information	88
	Define Enterprise: Manage Locations	89
	Define Geographies	91
	Define Legal Jurisdictions and Authorities	119
	Define Legal Entities: Manage Legal Entity	123
	Define Legal Entities: Manage Legal Entity HCM Information	127
	Define Legal Entities: Manage Legal Entity Tax Profile	134
	Define Legal Entities: Define Legal Reporting Units	137
	Define Business Units: Manage Business Units	138
	Define Business Units: Assign Business Unit Business Function	139
	Define Business Units: Manage Service Provider Relationships	141
	Define Business Units: Specify Customer Contract Management Business Function Properties	144
	Define Business Units: Specify Supplier Contract Management Business Function Properties	147
5	Define Security	151
	Define Security: Overview	151
6	Define Approval Management	153
	Approval Management: Highlights	153
	Other Workflow Setup	154
7	Define Help Configuration	157
	Setting Up Help: Overview	157
	Set Help Options	158
	FAQs for Assign Help Text Administration Duty	160
	Manage Help Security Groups	160



8 [	Define Application Toolkit Configuration	163
D	Define Application Toolkit Configuration: Overview	163
Ν	Map Reports to Work Areas	163
S	Set Watchlist Options	165
Ν	Manage Application Toolkit Administrator Profile Values	167
9 [	Define Applications Core Configuration	169
D	Define Applications Core Configuration: Overview	169
Ir	mporting Flexfields, Lookups, or Profile Values Using Web Services: Example	169
D	Define Lookups	172
Ν	Manage Messages	182
D	Define Document Sequences	188
D	Define Trees	191
D	Define Profile Options	203
D	Define Flexfields	208
D	Define Attachments	292
10 E	External Integration	297
V	Veb Services	297
F	Files for Import and Export	303
Е	External Data Integration Services for Oracle Cloud	305
11 li	mporting and Exporting Setup Data	317
C	Configuration Packages: Explained	317
lr	mplementation Project Based Export and Import: Explained	317
Ν	Moving Common Reference Objects	318





### Preface

This preface introduces information sources that can help you use the application.

### **Oracle Applications Help**

Use the help icon (?) to access Oracle Applications Help in the application. If you don't see any help icons on your page, click the Show Help icon (?) in the global header. Not all pages have help icons. You can also access Oracle Applications Help at https://fusionhelp.oracle.com.

### Using Applications Help

Watch: This video tutorial shows you how to find help and use help features.

### Additional Resources

- Community: Use Oracle Applications Customer Connect to get information from experts at Oracle, the partner community, and other users.
- Guides and Videos: Go to the Oracle Help Center to find guides and videos.
- Training: Take courses on Oracle Cloud from Oracle University.

### **Documentation Accessibility**

For information about Oracle's commitment to accessibility, see the Oracle Accessibility Program .

### Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send e-mail to: oracle\_fusion\_applications\_help\_ww\_grp@oracle.com.





### 1 Overview

### Common Implementation: Overview

Common implementation involves performing tasks that are common and apply to multiple offerings. The Define Common Applications Configuration, Define Transactional Business Intelligence, and Define Extensions task lists comprise these common setup and implementation tasks. You can search and access the common implementation task lists and tasks from the Setup and Maintenance work area.

### Define Common Applications Configuration

Within an offering, you may find the Define Common Applications Configuration task list containing several tasks to manage common functional areas, for example:

- Setting up security, enterprise structures, geographies, and business units.
- Creating and maintaining user accounts and synchronizing the list of users and roles stored in Lightweight Directory Access Protocol (LDAP).

### Define Transactional Business Intelligence

Use the Define Transactional Business Intelligence task list to configure business intelligence to gain real-time insight into transactional data. You can also secure the transactional data and manage the users accessing that data.

#### Define Extensions

Use the Define Extensions task list to configure offering-specific extensions to support customization of business objects. For example, the task list comprises tasks that help you to:

- Set options for the help features available at the site, which might include collaboration, access to external web sites, and settings for custom help content.
- Review and manage objects, for example currencies and reference data sets, that are shared across applications.
- Configure common reference objects such as flexfields, document sequences, and profile options that affect the functionality and look of Oracle Applications Cloud.

### Understanding Implementation Structures



### Setup and Maintenance: Overview

Oracle Functional Setup Manager enables rapid and efficient planning, configuration, implementation, deployment, and ongoing maintenance of Oracle Applications through self-service administration.

All Oracle Functional Setup Manager functionality is available from the Setup and Maintenance work area, which offers you the following benefits:

#### Self-Service Administration:

Manage all aspects of functional setup of Oracle Fusion applications at the business user level with an integrated, guided process for planning, configuration, implementation, deployment, and maintenance.

#### Configurable and Extensible:

Configure and Extend prepackaged list of tasks for setting up Oracle Fusion applications to better fit your business requirements.

#### Complete Transparency:

Get full visibility of Oracle Fusion applications end-to-end setup requirements with auto-generated, sequential task lists that include prerequisites and address dependencies.

#### Prepackaged Lists of Implementation Tasks:

Task lists can be easily configured and extended to better fit with business requirements. Autogenerated, sequential task lists include prerequisites and address dependencies to give full visibility to end-to-end setup requirements of Oracle Applications.

#### Rapid Start:

Specific implementations can become templates to facilitate reuse and rapid-start for comparable Oracle 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 Applications.

With Oracle Functional Setup Manager you can:

- Learn about and analyze implementation requirements.
- Configure Oracle Applications to match your business needs.
- Achieve complete visibility to set up 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 Applications through a standard and consistent process.

### 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 is the highest level grouping of Oracle Fusion Applications functionality. They include functional areas, and alternative business rules known as features.

### Functional Areas: Explained

A functional area is a grouping of functionality within an offering. It may be an optional piece of functionality that you may want to implement as part of an offering. Optional functional areas can be included or excluded from their parent offering. Functional areas may be hierarchical, and therefore may be subordinate to another functional area. An offering has at least one base or core functional area and may have one or more optional functional areas. Additionally, one or more or features may be associated to an offering. Base functional areas indicate the core functionality that you need to implement for the offering to be operational. Optional functional areas indicate optional functionality that you may or may not implement for an offering.

#### Common Functional Areas

Some core functionality essential to an offering such as setting the Initial Users or the Legal Structures may be shared across offerings. These are known as common functional areas and appear across offerings. Although most of the tasks associated to a common functional area are the same regardless of the offering you implement, there may be some offering-specific tasks.

In general once you implement a common functional area for a given offering, you won't need to repeat its implementation for the remaining offerings, however, it's recommended you check if there is any offering specific tasks that may still require your attention.

#### Base and Optional Functional Areas

Functional areas that support core functionality for an offering are known as base functional areas and must be implemented in order for the offering to be operational. Other functional areas known as optional functional areas support processes or functionality that can be implemented at your discretion depending on the business requirements. These can be implemented later during the implementation process.

### Features: Explained

Offerings include optional or alternative business rules or methods called feature choices, used to fine-tune business processes and activities supported by an offering or a functional area. You make feature selections according to your business requirements to get the best fit with the offering. If the selected offerings and functional areas have dependent features then those features are applicable when you implement the corresponding offering or functional area.

Feature choices can be one of three different types:

#### Yes or No.

If a feature can either be applicable or not be applicable to an implementation, a single check box is presented for selection. Check or deselect 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 check box. Select all that apply by checking the appropriate choices.

### Implementation Task Lists: Explained

The configuration of the offerings determine how the list of setup tasks is generated during the implementation phase. Only the setup tasks needed to implement the selected offerings, functional areas and features are included in the task list. This gives you the targeted task list necessary to meet your implementation requirements.

### Managing an Implementation

### **Enabling Offerings: Explained**

When planning your implementation, you decide what business processes your organization or company performs or supports. These decisions determine the offerings and functional areas you want to implement. You then configure the offerings and functional areas that support the activities your organization or company performs. During the configuration process, you specifically enable offerings and functional areas for use before you implement them.

#### **Enabling Offerings and Functional Areas**

Use the Setup and Maintenance work area to help decide which offerings to enable for implementation. Once you decide to use an offering, you can select the Configure button to choose the configuration details and enable the offering, associated functional areas, and features. All the base functional areas of an offering are automatically enabled for implementation when you enable the parent offering. You choose which optional functional areas to enable. The functional areas appear in an expandable and collapsible hierarchy to facilitate progressive decision making for implementation.

#### **Enabling Features**

Features are optional or alternative business rules or methods used to fine-tune business processes and activities supported by an offering or a functional area. If features are available for the offering or functional areas, you can enable them to help meet your business requirements, if desired. In general, the features are set with a default configuration based on their typical usage in most implementations. You should always review the available features for the offering and functional areas and select them as appropriate. Dependent features appear visible when the feature choice they depend on is selected for implementation.

### **Enabling Offerings: Procedure**

You enable offerings to customize the functionality that matches the services you plan on implementing.

### **Enabling Offerings**

To enable offerings, follow these steps.

- 1. Open the Setup and Maintenance work area (Navigator > Setup and Maintenance).
- 2. In the Setup and Maintenance Offerings page, select the offering you're using, then click **Configure**.



- 3. In the Configure page, select the **Enable** check box for the offering. Also select the **Enable** check box for each of the functional areas you want to use.
- 4. Click the Features icon for the offering or functional area you have enabled, then enable any features you require. Select **Done** when complete.
- 5. Select **Done** to return to the Offerings page then repeat the same steps for each of the offerings you are using.

### Implementing Offerings: Explained

Once you have configured the offering you want to implement, you can start performing the appropriate task to setting your applications up to support your business processes. Functional Setup Manager provides two methods to set up the offerings and therefore applications depending on your business needs.

#### Offering based implementation

Following a predefined list of tasks required for the features you selected to implement. This method enables you to implement the functionality on an adopt-as-you-go based approach. It provides you direct access to the setup tasks saving you time as by default gives you visibility to the minimum requirements for your implementation. This is always the recommended method to implement your applications unless you require custom implementation task lists.

#### Project based implementation

Enables you to customize your implementation defining an implementation project with a tailored list of tasks, task assignment and implementation progress monitoring. Use of this method is recommended when you require a custom task list.

### Offering Based Implementation: Explained

You can use the Setup and Maintenance work area to directly implement an entire offering or functional areas within an offering. You do not need to create an implementation project, and instead use a modular approach to your implementation. You can complete setup of specific business areas quickly to start transactions, and then gradually adopt more and more application functionality as needed.

An offering or functional area-based approach means you set up various parts of an offering at different times. You can start with set up of the functional areas that you immediately need to adopt. Over time, you can continue to set up other functional areas as you start to adopt additional applications functionality. Offerings must be enabled for implementation in order for their functional areas to display. Offering or functional area-based implementation provides the following advantages:

- When you select an offering the relevant functional areas appear for selection. The common functional areas are
  those shared across offerings and are listed first. The functional areas that are only associated with the selected
  offering, are at the bottom of the list.
- A functional area usually has several setup tasks, but only a few of them require input before the application function
  is ready for transactions. The rest of the setup tasks are usually optional or have predefined default values based on
  common use cases. When you select a functional area for implementation, you can view just the required tasks, or
  you can view the full list of setup tasks for the functional area.

### **Executing Setup Tasks**

You select the functional area you want to implement and the list of tasks that you need to perform appears. The tasks are organized with prerequisites and dependencies in mind. Select the task for which you want to enter data and then click Go to Task to render the page where you perform the task. If the setup data entered through a task can be segmented by a specific attribute, and therefore could be performed iteratively for each qualifying value, then the task may benefit from scope. Typical



examples include tasks relevant to legal entities, business units, ledgers, tax regimes, and legislative data roles. For such tasks, you are prompted to pick a scope value before entering data. You can pick a scope value that was previously selected, select a new scope value, or create a new scope value and then select it. The selected value is a qualifying attribute of the setup data entered by way of the task, and therefore, different setup data can be entered for different scope values. Enter data as appropriate and once you finish, close the page and you return to the functional area list of tasks.

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

### Project Based Implementation: Explained

You can create implementation projects to manage the implementation of an offering and functional areas as a unit throughout the implementation life cycle, or maintain the setup of specific business processes and activities customizing the list of tasks to complete their implementation.

An implementation project is the list of setup tasks you need to complete to implement selected offerings and functional areas. You create a project either by:

- selecting an offering and its functional areas you want to implement together, then customize the list of tasks for such offering and functional areas as applicable.
- selecting specific setup task lists and tasks you require for a specific configuration.

You can also assign these tasks to users and track their completion using the included project management tools.

#### Selecting Offerings

When creating an implementation project you see the list of offerings and functional areas that are configured for implementation. Implementation managers specify which of those offerings and functional areas to include in an implementation project. It is strongly recommended that you limit your selection to one offering per implementation project, even though the application does not prevent you from including more than one. The implementation manager should decide based on how they plan to manage their implementations. For example, if you implement and deploy different offerings at different times, then having separate implementation projects help to manage the implementation life cycles. Furthermore, the more offerings you included in an implementation project, the bigger the generated task list is. 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 and ensures that import and export sequence of the project data is straightforward in the correct sequence.



### 2 Maintain Common Reference Objects

### Maintain Common Reference Objects: Overview

The Maintain Common Reference Objects task list contains tasks that support implementation of common functionality, such as data security, reference data sets, or general preferences.

Use this task list to manage common reference objects that are defined centrally and shared across applications. You can search for and access this task list in the Setup and Maintenance work area.

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

#### Related Topics

• Moving Common Reference Objects: Overview

### Why can't I edit setup data?

The configuration of your setup data may be protected. Application developers mark some configurations as protected, indicating that you can't edit them.

Some examples of configurations that may be protected are:

- Descriptive flexfields
- Extensible flexfield contexts
- Extensible flexfield pages
- Value sets
- Tree structures

### **Define Application Taxonomy**

### Application Taxonomy: Highlights

Oracle application components and functions are organized in a hierarchy, ranging from product lines to logical business areas. The hierarchy represents a breakdown of products into units based on how applications are installed and supported.

In the Setup and Maintenance work area, search for the Manage Taxonomy Hierarchy task and view the hierarchy on the Manage Taxonomy Hierarchy page.

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 life cycle.

See: How to Manage the Life cycle

### Modules in Application Taxonomy: Explained

The top level of the hierarchy is product line, followed by the 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. A module is a node at any of these levels. Each level is briefly described here.

- Product Line: A collection of product under a single brand name, for example, Oracle Fusion.
- Product Family: 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: A single product within a product family, containing closely related features for a specific business solution, for example General Ledger.
- Logical Business Area: 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 up to two or three levels deep.

### Managing Modules in Application Taxonomy: Points to Consider

In the application taxonomy hierarchy, when you create a module, it becomes a child of the currently selected node. Once created, you cannot delete the module or move it elsewhere in the hierarchy.

From the Manage Taxonomy Hierarchy page, navigate to the Create Child Module or Edit Module page to manage the modules. 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, a unique read-only ID is automatically generated. The module contains two other identifiers: Module key and alternative 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 additional identifiers are provided for the product line, product family, and application modules. However, you can optionally add them for logical business areas and new custom modules.

Note: Don't 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 Oracle Application.

#### Names

Module name is the logical name for the module. The name must be unique among nodes within the hierarchy level with the same parent, but Oracle recommends keeping it unique in the entire hierarchy. The user name and description can appear to users in other parts of Oracle Applications Cloud.

#### **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 doesn't mean the module is actually installed if the installation itself didn't take place. Installation refers to operations related to laying down all the components required to create an Oracle Applications Cloud environment. 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 data residing in 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 the 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 Applications Cloud 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 common data is shared or partitioned across business entities to avoid duplication and reduce maintenance effort. 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.

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 can't 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 different business units setting up and using the same grades, XYZ Corporation decides to create a set called Grades, which contains the grades. All business units in the organization have the Grades set so that the grades can be shared and used.



Note: For specific information about configuring reference data sharing for a particular object or product, refer to the relevant product documentation.

### Reference Data Sets: Explained

Reference data sets are logical groups of reference data that various transactional entities can use depending on the business context. You can get started using either the common reference data set or the enterprise set depending on your implementation requirement. You can also create and maintain custom reference data sets, while continuing to use the common reference data set.

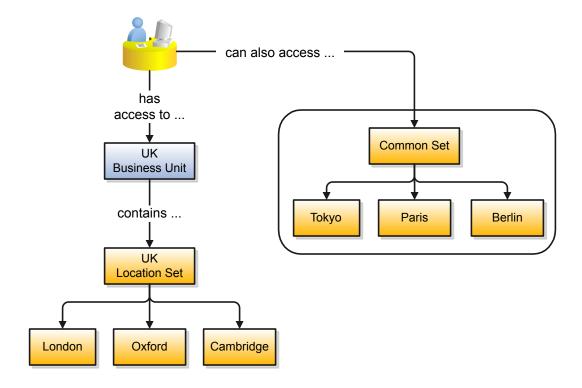
Consider the following scenario. Your enterprise can decide that only some aspects of corporate policy should affect all business units. The remaining aspects are at the discretion of the business unit manager to implement. This enables your enterprise to balance autonomy and control for each business unit. For example, your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level. Then, you can let managers define their own sales methods, but define payment terms centrally. As a result, each business unit has its own reference data set for sales methods and one central reference data set for payment terms assigned to all business units.

#### **Partitioning**

Partitioning reference data and creating data sets provide you the flexibility to handle the reference data to fulfill your business requirements. You can share modular information and data processing options among business units with ease. You can create separate sets and subsets for each business unit. Alternatively, you can create common sets or subsets to enable sharing reference data between several business units, without duplicating the reference data.



The following figure illustrates the reference data sharing method (assignment to one set only, with common values). 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.



#### Related Topics

Defining Default Reference Data Sets: Points to Consider

### 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. As a result, there is a reduction in the administrative burden and the time to implement new business units. For example, you can share sales methods, or transaction types across business units. You may also share certain 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 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 determine that certain aspects of your corporate policy can affect all business units. The remaining aspects are at the discretion of the business unit manager to implement. This allows your enterprise to balance autonomy and control for each business unit. For example, your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level. In such a case, you can let managers define their own sales methods, but define payment terms centrally. In this example:

- Each business unit has its own reference data set for sales methods.
- One central reference data set for payment terms is 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 set up 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

Variations exist 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. This method is 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. This method is 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. You need not explicitly
  assign 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 the following:
  - o Transaction types from the set assigned to the business unit.
  - 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 assign Net 15 to a set specific only to your business unit. At transaction entry, the list of values for payment terms consists of only 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.

#### Related Topics

- Items and Supplier Site Reference Data Sharing: Explained
- What reference data objects can be shared across cost organizations?
- What reference data objects can be shared across project units?
- What reference data objects can be shared across business units?
- What reference data objects can be shared across asset books?



# Assigning Reference Data Sets to Reference Objects: Points to Consider

You can assign the reference data sets to reference objects using 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 the 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 using a business context setting called the determinant type. A determinant type is the point of reference used in the data assignment process. The following table lists the determinant types used in the reference data assignment.

Туре	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 (also called determinant value) is a value that corresponds to the selected determinant type. The determinant is one of the criteria for selecting the appropriate reference data set.

### Reference Groups

A transactional entity may have multiple reference entities (generally considered to be setup data). However, all reference entities are treated alike because of similarity in implementing business policies and legal rules. Such reference entities in your application are grouped into logical units called reference groups. For example, all tables and views that define Sales Order Type details might be a part of the same reference group. Reference groups are predefined in the reference groups table.

### 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 can't change a currency code after you enable the currency, even if you later disable that currency.

#### **Date Ranges**

You can enter transactions denominated in the currency only for the dates within the specified range. If you don't enter a start date, then the currency is valid immediately. If you don't enter an end date, then the currency is valid indefinitely.

#### Symbols

Some applications support displaying currency symbols. You may enter the symbol associated with a currency so that it appears along with the amount.

### 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. If you have to use a different currency for Euro, you can disable the predefined 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 don't 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 must 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 can't be deleted, but 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 must 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?

The predefined territories are countries from the International Organization for Standardization (ISO) 3166 standard. Edit territory descriptions to determine how they are displayed in lists of country values in an application. You don't have to edit territory names or codes unless there is a specific requirement. Create territories if new countries emerge and the application isn't yet patched with the latest ISO country values.

Note: The National Language Support (NLS) territory codes are territory identifiers used in the application. Don't edit the codes unless you must change the association between ISO and the application territory.

#### When do I create or edit industries?

To meet a specific business need, you may edit industry names or descriptions of industries except for those belonging to the North American Industry Classification System (NAICS). Edit the industry descriptions also to determine how they appear in an application.

You may also create industries that contain customizations not included in the NAICS standards.

#### When do I associate industries with territories?

To meet specific business needs, you can associate industries with territories. For example, administrators can customize a page in different ways for different sets of users of the same industry, but residing in different countries.

#### When do I create or enable currencies?

Create or enable any currency for displaying monetary amounts, assigning currency to ledgers, entering transactions, recording balances, or for any reporting purpose. All currencies listed in the International Organization for Standardization (ISO) 4217 standard are supported.

The default currency is set to United States Dollar (USD).

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

Precision refers to the number of digits placed to the right of the decimal point used in regular currency transactions. For example, USD would have 2 as the precision value for transactional amounts, such as \$1.00.

Extended precision is the number of digits placed to the right of the decimal point and must be greater than or equal to the precision value. For calculations requiring greater precision, you can enter an extended precision value such as 3 or 4. That would result in the currency appearing as \$1.279 or \$1.2793.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for a cent.



In Setup and Maintenance work area, search for the Manage Currencies task to set these values for a currency.

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

The statistical unit currency type denotes the Statistical (STAT) currency used to record financial statistics in the financial reports, allocation formulas, and other calculations.

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

Edit the names and descriptions of International Organization for Standardization (ISO) languages to determine how they appear in the application. The ISO languages are a part of the ISO 639 standard. If any change to the ISO standard doesn't reflect in the application, you can update the ISO alpha-2 code or add languages to provide up-to-date information.

#### When do I edit languages?

Installed languages automatically appear on the Manage Languages page. This page also displays all languages that are available for installation and translation. Each dialect is treated as a separate language.

Generally, you don't need to edit any of the detailed fields unless absolutely necessary.

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

Though all standard time zones are provided, enable only a subset for use in lists of time zone values. You can add time zones if new zones became standard and the application isn't yet patched with the latest values.

### 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 required for reporting.

### **Enabling Audit Functionality**

For Oracle Applications Cloud, 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 and manage audit, ensure that you have a role with the assigned privilege Manage Audit Policies (FND\_MANAGE\_AUDIT\_POLICIES\_PRIV). For appropriate assignment of roles and privileges, check with your security administrator.

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, see Audit Events for Oracle Applications Cloud Middleware (Doc ID 2114143.1) on My Oracle Support at https://support.oracle.com.

If you don't want an application to be audited, you can stop the audit process by setting the Audit Level option to None.



#### Related Topics

Audit Events for Oracle Applications Cloud Middleware

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

Audit enables tracking 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 operations performed on an object and its attributes, such as create, update, and delete. To configure audit business object attributes, navigate to the Manage Audit Policies page in the Setup and Maintenance work area.

#### Selecting an Application

To set up auditing, you must select a web application that contains the required business objects that can be audited. From the list of business objects, select those business objects 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. If the object selected in an audit hierarchy is also a part of several other audit hierarchies, the attribute configuration for that object is applicable to all the hierarchies in that application.



🔽 Tip: For business objects based on flexfields, select the Flexfields (Additional Attributes) check box to view and add or remove flexfield attributes, to include or exclude them from the audit.

#### 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 Applications Cloud 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 users view the audit history for an application, they can specify the period for which they want the results. Therefore, make a note of when you start and stop auditing an application.

For example, users intend to view the audit history of an object for the previous week, but auditing for that object was stopped last month. They wouldn't get any audit results for that week, because during the entire month that object wasn't audited. Even if you enable audit for that object today, users can't get the wanted results because audit data until today isn't available.

### Configuring Audit: Highlights

To set up auditing for Oracle Applications Cloud, use the Manage Audit Policies page in the Setup and Maintenance work area. To set up auditing for Oracle Fusion Middleware products, select the level of auditing mapped to a predefined set of metadata and the events that have to be audited. Information about 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 as audit-specific assets that you can use to create the config.xml configuration file. To get a list of audit-specific assets, see Audit Events for Oracle Applications Cloud Middleware (Doc ID 2114143.1) on My Oracle Support at https://support.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

#### Related Topics

Audit Events for Oracle Applications Cloud Middleware

### Audit Impersonation: Explained

Users can temporarily designate other users to impersonate their profiles and perform application tasks on their behalf. By default, all impersonations are audited. In the global area, click your user name and from the Settings and Actions menu, select **Set Preferences**.

At run time, audit setup tracks and stores information about all attributes, whether auditing is enabled for an attribute or not. As a result, impersonation auditing is also active even when auditing is disabled for an application. The audit history captures the impersonator information in addition to the actual user information. Therefore, while viewing audit history, users can retrieve the audited information, filtered by an impersonated user.

To assign or modify impersonations, in the Tasks pane on the Preferences page, click **Proxies**. You can search for the users who can be impersonated and switch the access to that user.

Note: The associated profile option Audit Impersonation Transaction Enabled is enabled by default. To disable it, set its profile value to **No**.

#### Related Topics

Proxies: Explained

### Manage Oracle Social Network Objects



### Managing Oracle Social Network Objects: Explained

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

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

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

To open the Manage Oracle Social Network Objects page, start in the **Setup and Maintenance Overview** page and search for the Manage Oracle Social Network Objects task.

Use Oracle Social Network to:

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

Note: Oracle Social Network is currently available in Cloud implementations only.

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

### **Enabling Business Objects for Integration**

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

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

### Enabling Social Networking on Objects: Critical Choices

You can determine whether information about a business object, such as benefit plans or sales accounts, displays in Oracle Social Network. If you enable an object for sharing, you allow users to collaborate on the object through social networking. You can choose whether all instances of an object are shared, or only at the user's discretion. You can also choose which attributes are shared, such as names, details, and who made the last update.

In addition to a wide range of predefined objects, you can share:

Custom objects and attributes created in Application Composer



· Custom fields created in descriptive flexfields

Use the Manage Oracle Social Network Objects task in the Setup and Maintenance work area.

After you click **Enable Object**, select one of the following enablement options:

- Manual
- Automatic
- No

#### Manual

If you select this option, which is recommended, you let users decide whether to share each instance of the object with the social network. Once shared, all updates to the enabled attributes of the instance appear on the social network. If the instance is deleted, that information is also shared.

Click **Enable All** to enable all objects for all applications. Enable All automatically applies the Manual option, which means that the user can choose whether to share an object instance.

#### **Automatic**

With this option, news about all instances of the object appears on the social network, including:

- Every newly created instance
- All subsequent updates to the enabled attributes
- Deletion of any instances

#### No

With this option, which is the default value, no news about the object appears on the social network.

Note: When you click **Disable Object**, the enabled setting of the selected business object is automatically changed to No.

After you enable a business object, you must enable one or more attributes of the object. Only the enabled attributes are shared. The Status column in the Business Objects table indicates which enabled business objects don't yet have an enabled attribute. For these objects, only the following information appear on the social network:

- Internal bookkeeping information, when creating or updating an instance of the object.
- · News that an instance is deleted.

### Update Translations: Explained

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

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



### Synchronize Business Objects: Explained

Use **Synchronize** on the Manage Oracle Social Network Objects page to synchronize business objects. This resends the definitions of business objects having the enablement option as **Manual** or **Automatic** to Oracle Social Network.

Use the Synchronize button at the:

- Business Objects table level: To resend the definitions of a selected business object to social network. This
  button is enabled only when you select a row for a business object with the enablement option as Manual or
  Automatic.
- Manage Oracle Social Network Objects page level: To resend the definitions of all business objects with the
  enablement option as Manual or Automatic to social network.
- Note: If you had modified any business object enabled for social network and not saved your changes, then on clicking **Synchronize**, a warning message appears. This message informs you that you have not saved your changes, and you can select one of the following options:
  - Save and Synchronize: To save the modified business objects, and synchronize the unmodified business objects.
  - Synchronize: To ignore any unsaved business objects, and only synchronize the unmodified business objects.
  - Cancel: To cancel the synchronization task.

### FAQs for Manage Oracle Social Network Objects

#### What happens if I update translations?

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

On updating translations, you also:

- Synchronize the newly translated text from Oracle Applications Cloud so that it can be used within social network. This means you can:
  - o Install and enable a new language.
  - Take a language patch at any time.
- Send attribute labels and business object names to social network for use in its user interface.

### How can I update translations?

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

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

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



- Manage Oracle Social Network Objects page level: To send translations for all business objects with the enablement option as Manual or Automatic to social network.
- Note: When you save the enablement of a business object to social network, it sends the translations as well. Hence, you need not click **Update Translations** after saving the enablement.

#### When do I update translations?

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

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

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

#### What happens if I synchronize business objects?

When you synchronize business objects, you resend the definitions of business objects having the enablement option as **Manual** or **Automatic** to Oracle Social Network.

#### When do I synchronize business objects?

Run the Synchronize process after you use customization sets to import the setup from the Manage Oracle Social Network Objects page in another environment.

You can also run the process whenever you want to synchronize the settings of business objects with social network without making changes in the Manage Oracle Social Network Objects page.

#### Related Topics

Using Customization Migration to Move Customizations: Points to Consider

### Manage Applications Core Common Reference Objects

### Manage Applications Core Messages

### Common Messages: Points to Consider

Message names that begin with FND\_CMN are common messages. Each common message can appear in multiple places in any product family across Oracle Fusion Applications. For example, the FND\_CMN\_NEW\_SRCH message can be used for any search to indicate that no results were found. Common messages of type error or warning are part of the message dictionary.

#### Creating and Editing Common Messages

You can create custom common messages for use in multiple places. However, ensure that you follow the predefined naming convention and numbering series associated with the application or module.



Note: Don't use FND\_CMN as the prefix for your custom messages because all the predefined common messages begin with it.

Common messages can be used in any application. Therefore, 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, ensure that the text is generic and applies to the entire site of Oracle Fusion Applications implementation.

### Manage Applications Core Administrator Profile Values

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

The general preferences such as Language, Territory, or Date Format that you access from the global area have corresponding predefined profile options.

#### General Preferences

When users define their preferred Date Format, Language, or Currency, they are setting the value of a profile option at the user level.

#### **Profile Options**

When users don't specify anything as their preferences, the Site level profile option takes effect.

#### Setting Up for General Troubleshooting: Points to Consider

To help the help desk troubleshoot issues that users encounter in the application, users can record the issue while they reproduce it. Some advanced users might also need detailed information in the About This Page dialog box. Setting up for troubleshooting involves making sure that users have the right access, and determining how many users can record at the same time.

#### Access

Check with your security administrator that the appropriate users are assigned roles that inherit the following privileges:

- Record and View Issue (FND\_RECORD\_AND\_VIEW\_ISSUE\_PRIV): To create a basic recording
- Set Issue Recording Advanced Options (FND\_SET\_ISSUE\_RECORDING\_ADVANCED\_OPTIONS\_PRIV): To set advanced options before starting the recording
- View Version Information (FND\_VIEW\_VERSION\_INFORMATION\_PRIV): To see the versions that technical
  components of the application are on

#### Number of Users

Recordings are stored on servers, and by default, up to five users can record at the same time on each server. For performance reasons, you can set the Maximum Number of Users Allowed to Record Issues (ORA FND RECORD ISSUE MAX USERS) profile option to a number lower than five.

#### Related Topics

- Recording Issues to Troubleshoot: Procedure
- How can I view the version information of an application?
- Setting Profile Option Values: Procedure



#### Managing Profile Option Values for CORS Headers: Points to Consider

You can set profile option values for the CORS headers using the Manage Administrator Profile Values task in the Setup and Maintenance work area.

#### **CORS Headers**

This table lists the CORS headers that you can set profile option values for.

CORS Header	Profile Option Name (Profile Option Code)	Profile Option Values
Access-Control-Allow-Origin	Allowed Domains (ORACLE. ADF. VIEW. ALLOWEDORIGINS)	<ul> <li>Valid values for allowed origins:</li> <li>URL of the specific origin, for example, http://www.mydomain.com</li> <li>Comma-separated list of origins, for example, http://www.mydomain.com, http://adc6160507.us.oracle.com, http://software.dzhuvinov.com</li> <li>* to allow access to resources from all origins</li> <li>Empty (no value set) to prevent access to resources from any origin</li> <li>Note: You must set a value for this header to enable CORS.</li> </ul>
Access-Control-Max-Age	CORS: Access-Control-Max-Age (CORS_ACCESS_CONTROL_MAX_AGE)	Default value for caching preflight request is 3600 seconds.
Access-Control-Allow-Methods	CORS: Access-Control-Allow-Methods (CORS_ ACCESS_ CONTROL_ ALLOW_ METHODS)	Default values for allowed methods are OPTIONS, HEAD, GET, POST, PUT, PATCH, DELETE.
Access-Control-Allow-Headers	CORS: Access-Control-Allow-Headers (CORS_ ACCESS_ CONTROL_ ALLOW_ HEADERS)	Default values for allowed headers are Accept, Accept-Encoding, Cache-Control, Content-MD5, Content-Type, If-Match, If-None-Match,Origin, User-Agent, X-HTTP-Method-Override, X-Requested-By.
		Note: You must include Authorization, with a comma as the delimiter, to the list of allowed headers. For example: Accept, Accept- Encoding, Cache-Control, Authorization
Access-Control-Allow-Credentials	CORS: Access-Control-Allow-Credentials (CORS_ ACCESS_ CONTROL_ ALLOW_ CREDENTIALS)	True to enable sending credentials with the request



CORS Header	Profile Option Name (Profile Option Code)	Profile Option Values
		False, which is the default value, to disable sending credentials with the request

#### Related Topics

- Setting Profile Option Values: Procedure
- CORS: Explained

#### How can I enable the privacy statement?

In the Setup and Maintenance work area, open the Manage Applications Core Administrator Profile Values task and search for the **Privacy Statement URL** profile option. In the profile values section, update the **Profile Value** text box with the full URL of the web page containing the privacy content.

In the global area, click your user name and from the Settings and Actions menu, select **About This Page.**. Click **Privacy Statement** to view the linked web page.

#### How can I make message components visible only to specific users?

Use the Manage Administrator Profile Values task to determine the visibility of the message components. For the **Message Mode** profile option, set the profile value to either User or Administrator. Based on the set value, the administrator or user actions and details appear for the intended audience.

However, the message components are visible to the audience based on their predefined access rights. Anyone having a user level access can't view the Administrator message components. If you set the profile value to the administrators of a specific product, the message components are visible only to that specific audience.

Note: If you don't set any value to the profile option, the visibility of the message component is determined by the default profile option settings.

#### Related Topics

• Setting Profile Option Values: Procedure

### Define Global Search

### Setting Up the Global Search: Overview

You have many options for setting up the global search, from enabling or disabling it completely to controlling what appears in the autosuggest. For most of the steps, use setup tasks in the Setup and Maintenance work area.

### **Enabling Global Search**

A predefined set of business objects, for example help files in Applications Help, are available for global search. These objects are grouped into categories. The search gives you results from indexed data, for example the help files with text that matches your search term.



Enabling global search includes setting up the predefined objects and categories, and starting a schedule to refresh the index so that users get the latest results.

- 1. Check with your help desk to make sure that objects, categories, and schedules are already set up.
  - If you're not using an Oracle Cloud implementation, you need to do this setup yourself. For details about enabling search for predefined objects, categories, and indexes, see the Oracle Fusion Applications Administrator's Guide.
- 2. Set the Global Search Enabled (FUSION\_APPS\_SEARCH\_ENABLED) profile option to Yes at the Site level.
  - o Otherwise, the search field isn't available in the global area for any user.
  - o After you set the profile option, users must sign out and sign back in to see the global search.
- Note: Enabling global search is the only mandatory setup. You can skip the rest of the steps below, and just use the default configuration for the global search.

### Defining Global Search

Use these tasks in the following tasks lists to control how the global search works:

- Define Global Search task list
  - Manage Global Search Configurations: Define configurations that capture a collection of settings, and determine which configurations are enabled or used as default. Each configuration applies to the global search on all or specific applications or pages.
  - Manage Suggestion Groups: Define suggestion groups, which represent categories of suggestions in the autosuggest.
- Manage Applications Core Common Reference Objects task list
  - **Manage Applications Core Alternate Words:** Define alternates for search terms that users enter, so that the search can also find matches based on the alternate terms.

### Retaining Recent Items

You can set the Recent Items Retention (FND\_PURGE\_RECENT\_ITEMS) profile option to determine how long to retain recent items for the global search autosuggest. A page that the user opened can appear in the user's autosuggest until the specified number of days passes. Aside from the autosuggest, this profile option also applies to the recent items tracked under **Favorites and Recent Items** in the global area.

#### Related Topics

• Setting Profile Option Values: Procedure

### Manage Global Search Configurations

#### Setting Global Search Configurations as Enabled or Default: Examples

Each global search configuration contains settings for the global search, and a configuration can apply to specific pages or applications. Use the Manage Global Search Configurations page to enable or disable configurations, and select the one to use as the default. The following scenarios explain which configurations actually apply to the global search, depending on the configurations that you enable or set as default.



#### **Predefined Default**

The predefined Default configuration is always enabled and set as a default. This configuration is the working default unless a custom global search configuration is also set as a default. In this scenario, you don't enable any other configuration, so this Default configuration applies to the global search on all pages, in all applications.

#### **Custom Default**

You create a custom global search configuration that applies to page A and application B. Later, you set your configuration as the default. Only this custom configuration and the predefined Default configuration are enabled. Both are set as default.

The result is that:

- Your custom configuration overrides the predefined Default one and becomes the working default.
- Even though you defined your custom configuration to apply to page A and application B, it now actually applies to all pages and all applications. This is because your configuration is the working default, and no other configuration is enabled.

#### Specific Pages or Applications

You're using either the predefined Default configuration or a custom configuration as the default. You also enable:

- Configuration 1: Applies to application A
- Configuration 2: Applies to application B and a few pages in application A

The result is that:

- Configuration 1 applies to all pages in application A, except the few pages that use configuration 2.
- Configuration 2 applies to all pages in application B, plus the few pages in application A.
- The default configuration applies to all other applications and pages.

#### Creating Global Search Configurations: Procedure

Predefined global search configurations control how the global search behaves and looks. You can't edit these configurations, but you can duplicate them and edit the copies, or create your own from scratch.

#### Creating a Configuration

Follow these steps:

- 1. Open the Setup and Maintenance work area, and go to the Manage Global Search Configurations task.
- 2. Click **Create**, or select a row and click **Duplicate**.
  - Note: You can't delete a configuration after you create it, but you can disable it.
- 3. For the short name (identifier for your configuration), enter an alphanumeric code with uppercase letters and no spaces
- **4.** Enter a user-friendly name and description for the configuration.
- 5. Select the **Default** check box if you want to use your configuration as the default instead of the predefined Default configuration. If another custom configuration was already set as the default, then your configuration becomes the new custom default.
- **6.** Select a product family if the configuration is for applications or pages within a specific family. Otherwise, select **Common**.
- 7. If you're creating a duplicate, click **Save and Close**. To go on to the next steps and define more settings, select your configuration and click **Edit**.



- 8. Enter a module within the product family you selected. If you selected the Common family, then select the **Oracle Middleware Extensions for Applications** module.
- 9. Use the tabs to define your configuration:
  - Autosuggest: Determine what's available to users in the global search autosuggest, as well as how the autosuggest looks and behaves.
  - o Search Field: Control the search field in the global area and in the search results dialog box.
  - Search Results: Enable or disable saved and recent searches, select the search categories available to users, and define settings for filters.
  - Pages: Indicate the applications or pages that this global search configuration applies to.
- 10. Save your work.

#### Setting Up the Autosuggest for the Global Search: Procedure

Use global search configurations to determine what's available to users in the autosuggest. You select the suggestion groups to include in configurations. The configurations also determine how the autosuggest looks and behaves.

#### Prerequisite

Open the Autosuggest tab in the Create or Edit Global Search Configuration page.

#### Defining the Content

To select suggestion groups and determine how they're displayed in the autosuggest:

- 1. In the Suggestion Group section on the Autosuggest tab, move the groups you want to include into the **Selected Groups** pane.
  - The **Enabled** column in the Available Groups pane indicates if the group is defined (on the Manage Suggestion Groups page) to be displayed by default or not in the autosuggest.
- 2. In the **Enabled** column in the Selected Groups pane, select one of the following values. The **Displayed by Default** column shows the resulting behavior in the autosuggest, based on what you select in the **Enabled** column.

#### Note:

- Inherit: In the autosuggest, the group is displayed or hidden by default depending on what's defined for the group.
- Yes: The group is displayed by default, no matter what's defined for the group.
- No: The group is hidden by default, no matter what's defined for the group.
- 3. Order the selected groups as you want them to appear in the autosuggest.
- 4. Above the Suggestion Groups section, select the Enable personalization of search groups check box if you want to allow users to override your configuration. Users can hide, show, and reorder suggestion groups for their autosuggest.
- Tip: Click the **Manage Suggestion Groups** button at any time to edit or create suggestion groups. When you return to the Autosuggest tab, click **Refresh** to reflect the changes you made to suggestion groups.



### Defining the Appearance

Optionally define settings in the Appearance section on the Autosuggest tab:

- Show Suggestion Group Headings: Select this option to display suggestion group headings (text and icon) in the autosuggest. Even if you do so, if a group is defined on the Manage Suggestion Groups page to not show headings, then its heading won't be displayed.
- Show Icons: Select this option to display icons next to suggestions in the autosuggest.
- **No Suggestions Message:** Enter the message that appears when no suggestions match the user's search term. If you leave this field blank, then no autosuggest or anything at all appears when there are no matches.

### Defining the Behavior

Optionally define settings in the Behavior section on the Autosuggest tab:

- **Show Top Suggestions:** Enable this option to display suggestions in the autosuggest as soon as the user clicks in the search field, even without entering a search term. For example, the last few pages the user opened would appear as suggestions under the Recent Items group.
- **Minimum Characters for Autosuggest:** Enter the number of characters that users must enter in the search field before matching suggestions appear in the autosuggest.
- **Maximum Number of Suggestions:** Enter the maximum number of suggestions to be displayed across all suggestion groups. This total is distributed as equally as possible among the groups.

# Disabling Saved Searches and Recent Searches for the Global Search: Points to Consider

Global search configurations determine if saved searches and recent searches are enabled in the global search. Consider the following points when you use disable either. In the Create or Edit Global Search Configuration page, open the Search Results tab and use the Saved and Recent Searches section.

#### Disabling Saved Searches

If you disable saved searches:

- You disable the Save button in the search results dialog box, so users can't create or edit saved searches for global search.
- You're not disabling the Saved Searches suggestion group. Users can still see any applicable saved searches in the global search autosuggest.

#### Disabling Recent Searches

If you disable recent searches:

- The application isn't saving recent searches.
- You're not disabling the Recent Searches suggestion group. Users can still see in the autosuggest any applicable searches that they recently ran before you disabled recent searches.

### Setting Up Filters for the Global Search: Procedure

To determine how users can filter their search results, define the appropriate settings when you create or edit global search configurations. You can set up search categories so that users can limit the scope of their search to begin with, or refine their search results.

#### Prerequisite

Open the Search Results tab on the Create or Edit Global Search Configuration page.



### Setting Up Categories to Narrow Search Scope

To let users select the categories to search on before running the search:

- 1. In the Filters section, enable personalization of search categories.
- 2. In the Search Categories section, select the categories that users can search on. If you don't select any, then every category is available to users.

### Setting Up Categories as Search Result Filters

To let users filter search results based on category:

- 1. In the Filters section, select the check boxes to show subcategories, facets, or both. Categories are always displayed. Subcategories are an additional level of filters below categories, and facets are a level below categories.
- 2. Select a filter display style so that the list of all available category names are displayed:
  - o Inline: In the Filters pane in the search results
  - LOV: In a Categories dialog box that users can open from the Filters pane
- 3. In the Search Categories section, select the categories to use as filters. This is the same set of categories to be used for personalization. If you don't select any, then every category is available to users.

### Setting Other Options for Filters

You can also use the Filters section to:

- Show Hit Counts: Show the number of search results that match each filter value
- Enable Clear All Filters: Allow users to clear all filters with one click of a button

In the Last Updated Date Filters section, select the criteria to use as filters, based on the last update date. If you don't select any, then every date filter is available to users.

# Specifying the Pages or Applications That a Global Search Configuration Applies To: Procedure

As part of defining your global search configuration, you can specify the pages or applications (or both) that your configuration applies to. If you want your configuration to apply to all pages in all applications, then skip these steps.

### Prerequisites

If you want the global search configuration to apply to specific applications, you need to find the application short name.

- 1. Open the Setup and Maintenance work area and go to the Manage Taxonomy Hierarchy task.
- 2. Expand the Oracle Fusion node.
- 3. Select the row (with the Application module type) for your application, and click **Edit Module**.
- **4.** In the Application Details section, see the **Application Short Name** column and note down the value to use as the application short name.

### Adding Pages or Applications

Follow these steps:

- 1. On the Create or Edit Global Search Configurations page, open the Pages tab.
- 2. Click the Create icon.
- 3. In the View Type column, indicate if the configuration applies to a specific page or application.
- **4.** Enter a view ID to identify the page or application:



- Tip: You can use % as a wildcard for the page or application value, such as **Example**% for all pages that start with **Example**.
- Page: Enter the last part of the URL you get when you open that page. For example, enter ExamplePage from the URL http://exampleServer/homePage/faces/ExamplePage.
- Application: Enter the application short name with a wildcard at the end, for example номерадельря.
- 5. Add more pages or applications as needed.

# Manage Suggestion Groups

### Suggestion Groups for the Global Search: Explained

A suggestion group is a category of suggestions that users see in the autosuggest for the global search. For example, if the user enters Report in the search field, then the Navigator suggestion group in the autosuggest shows any Navigator menu names with Report.

### Managing Suggestion Groups

Each suggestion group can have a heading, for example Recent Items, as well as an icon that appears before the heading text. The icon helps users identify what the group is all about.

You can manage suggestion groups to:

- Show or hide the group by default in the autosuggest
- Enter the heading text
- · Define if the heading text appears in the autosuggest or not
- Identify the image file to use as the icon
- Tip: You can copy predefined suggestion groups and edit them to create customized versions.

To manage suggestion groups, open the Setup and Maintenance work area and use either the:

- Manage Suggestion Groups task
- Autosuggest tab when you create or edit global search configurations using the Manage Global Search Configurations task

### Using in Global Search Configurations

Just because a suggestion group is defined to be displayed by default doesn't necessarily mean that it in fact appears in the autosuggest. Global search configurations determine which groups are included for the autosuggest, whether icons appear for the headings, and so on.

You can use a suggestion group in many or all global search configurations. And ultimately, if personalizing suggestion groups is enabled, then users can show, hide, and reorder the suggestion groups included in the global search configuration.

### Changing the Heading Text and Icon for Suggestion Groups: Worked Example

This example shows how to change the icon and text for a suggestion group heading in the global search autosuggest. In this example, you start out with the predefined Default global search configuration enabled, and no custom configurations.



The following table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
Which predefined suggestion group do you want to change?	Recent Items
What do you want the new heading text to be?	Recently Visited Pages
Which image do you want to use as the icon?	A .png file (16 by 16 pixels) that's used on your company Web site
Do you want the new suggestion group to appear in the autosuggest by default?	Yes
Which pages should the changes apply to?	All pages

To use a new icon and heading text for the suggestion group:

- Make a copy of the predefined Recent Items suggestion group.
- Update global search configurations to use the custom suggestion group.

### Duplicating the Predefined Suggestion Group

- 1. Open the Setup and Maintenance work area, and go to the Manage Suggestion Groups task.
- 2. Select the Recent Items group and click **Duplicate**.
- 3. In the new row, enter **RECENTCUSTOM** as the short name.
- 4. Change the display name to **Custom Recent Items**.
- 5. Change the description to **Custom version of Recent Items**.
- 6. With your new row still selected, click Edit.
- 7. In the Heading section, enter **Recently Visited Pages** in the **Text** field.
- 8. In the **Icon** field, enter the full URL to your .png file.
- 9. Click Save and Close.

### Updating Global Search Configurations

- 1. In the Setup and Maintenance work area, go to the Manage Global Search Configurations task.
- 2. Select the Default configuration and click **Duplicate**.
- 3. Fill out the row for your new configuration, selecting the **Default** check box.
- 4. With the row still selected, click the Edit icon.
- 5. In the Autosuggest tab, click the **Refresh** button in the Suggestion Group section if you don't see your custom suggestion group.
- 6. Move the Custom Recent Items group into the **Selected Groups** list, and move the Recent Items group out.
- 7. In the **Enabled** column for the Custom Recent Items Group, select **Yes**.
- 8. In the Appearance section, make sure that headings are set to be displayed.
- 9. Click Save and Close.

### Creating Suggestion Groups for the Global Search: Procedure

For the autosuggest in the global search, you can create new suggestion groups that determine suggestions in a way that's different from the predefined groups. Creating such suggestion groups involves developer tasks and is not applicable to Oracle Cloud implementations. After the code for the group is ready, you define the new suggestion group in the Create Suggestion Group page.



Note: You can't delete a suggestion group after you create it.

### Prerequisites

Before you define the new suggestion group:

- Your developer writes code for the new suggestion group and provides you values for some of the suggestion group settings. For more information on implementing autosuggest, see the Oracle Fusion Applications Developer's Guide.
- If you want to display an icon for the group in the autosuggest, make sure that the graphics file:
  - Can be accessed using a URL.
  - o Is 16 by 16 pixels or smaller.

### Creating Suggestion Groups

Follow these steps:

- 1. Open the Setup and Maintenance work area and open the Manage Suggestion Groups task.
- Click Create
- 3. For the short name (identifier for your group), enter a unique alphanumeric code with uppercase letters and no spaces.
- 4. Enter a display name for the group, which can be different from the heading that appears in autosuggest.
- 5. Enter a description.
- 6. Determine if the suggestion group should be displayed by default in the autosuggest.
- 7. Select a product family if the group is for business objects from a specific family. Otherwise, use **Common**.
- 8. Select a module within the product family. If the product family is Common, then use the Oracle Middleware Extensions for Applications module.
- **9.** In the **Data Source** field, enter the value your developer provides to determine the records to display in this suggestion group.
- 10. In the Context Code and Object Type fields, enter the value your developer provides, if any.
- 11. In the Heading section, determine if the heading (text and icon) is visible in the autosuggest.
- 12. Enter the exact text to appear as the heading of this group in the autosuggest.
- **13.** To display an icon, enter in the **Icon** field the full URL to the image file, or a relative URL that your developer provides if the file is stored with other application artifacts.
- **14.** Save your changes.
  - Note: Global search configurations:
    - Determine which suggestion groups are actually available for the global search.
  - Determine whether each included group is visible by default or not in the autosuggest.
  - Can hide suggestion group headings in the autosuggest, even if the group is defined to show headings.

# Managing Alternate Words for Global Search: Points to Consider

Use the Manage Applications Core Alternate Words task in the Setup and Maintenance work area to maintain a list of search terms that users might use for the global search. For each user keyword, define a possible alternate to also search on. Consider various reasons for defining these word pairs, and also decide whether to automatically search on both terms or not.



### Reasons for Word Pairs

This table provides some reasons and examples for managing alternate words.

Purpose	What You Enter	User Input Keyword Example	Alternate Keyword Example
Correct user typos	A possible typo as the user input keyword, and the correctly spelled term as the alternate keyword	Oracel	Oracle
Account for abbreviations and acronyms	An abbreviation or acronym and what it stands for	ВІ	business intelligence
Account for common variations in spelling	Two different ways to spell the same term	email	e-mail
Enable matches on synonyms	A pair of terms that are functionally similar	hyperlink	link
Help new users who are not familiar with what things are called in the application	The term that your users might search on as the user input keyword, and the equivalent in the application as the alternate keyword	concurrent program	scheduled process

Note: You don't have to account for plurals or case sensitivity. For example, if you have **email** as a user input word, you don't have to also add **Email** or **emails** as an input word.

### Automatically Search Both Terms

For each pair of terms, use the Automatically Search Both check box to determine what happens when the user enters the input word and starts the search:

- Yes: The search runs and displays results based on both terms.
  - Before you select this check box, carefully consider possible impact. For example, would users get a lot of unnecessary search results, making it harder to find what they want?
- **No:** The user sees a message and can decide to continue searching on just the input term, or to search on just the alternate term instead.



# 3 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 can't change a currency code after you enable the currency, even if you later disable that currency.

### **Date Ranges**

You can enter transactions denominated in the currency only for the dates within the specified range. If you don't enter a start date, then the currency is valid immediately. If you don't enter an end date, then the currency is valid indefinitely.

### Symbols

Some applications support displaying currency symbols. You may enter the symbol associated with a currency so that it appears along with the amount.

#### Related Topics

- What's the difference between precision, extended precision, and minimum accountable unit for a currency?
- What's a statistical unit currency type?
- Euro Currency Derivation: Explained

# 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. If you have to use a different currency for Euro, you can disable the predefined 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 don't 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 must 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 or enable any currency for displaying monetary amounts, assigning currency to ledgers, entering transactions, recording balances, or for any reporting purpose. All currencies listed in the International Organization for Standardization (ISO) 4217 standard are supported.

The default currency is set to United States Dollar (USD).

#### Related Topics

Defining Currencies: Points to Consider

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

Precision refers to the number of digits placed to the right of the decimal point used in regular currency transactions. For example, USD would have 2 as the precision value for transactional amounts, such as \$1.00.

Extended precision is the number of digits placed to the right of the decimal point and must be greater than or equal to the precision value. For calculations requiring greater precision, you can enter an extended precision value such as 3 or 4. That would result in the currency appearing as \$1.279 or \$1.2793.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for a cent.

In Setup and Maintenance work area, search for the Manage Currencies task to set these values for a currency.

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

The statistical unit currency type denotes the Statistical (STAT) currency used to record financial statistics in the 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
- Fixed

You can 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

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

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

### 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 your 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.

#### Related Topics

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

# Using Rate Types: Examples

The four predefined conversion rate types in Oracle Fusion Applications are:

- Spot
- Corporate
- User
- Fixed



#### Scenario

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

- Canadian dollar (CAD): A 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 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.

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 (EUR) was 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 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 setting 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.



Rate Type	Usage
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. The user rate can also provide an accurate conversion rate on the date of the transaction.



# 4 Define Enterprise Structures

# 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, that describe its operations and provide a basis for reporting.

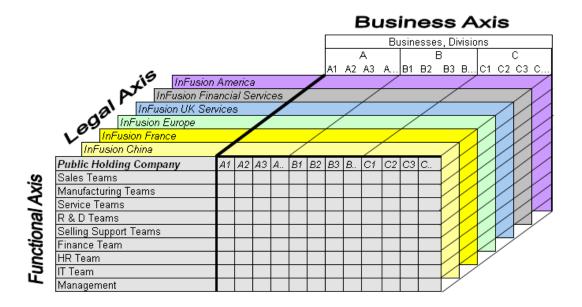
- Legal
- Managerial
- Functional

In Oracle Fusion, these structures are implemented using the chart of accounts and organization hierarchies. 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
- Departments



Aligned these structures with 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.
- Owned by its shareholders, who may be individuals or other corporations.

Many other kinds of legal entities exist, 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 the entity 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 contributes to 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 its 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 have to be reflected directly in the legal structure of the enterprise. The management structure can include divisions, subdivisions, lines of business, strategic business units, profit, 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 as well as being reflected in the chart of accounts.

### **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 and selling, general, and administrative expenses. In Oracle Fusion Applications, the functional structure is implemented using departments and organizations, including sales, marketing, project, cost, and inventory organizations.

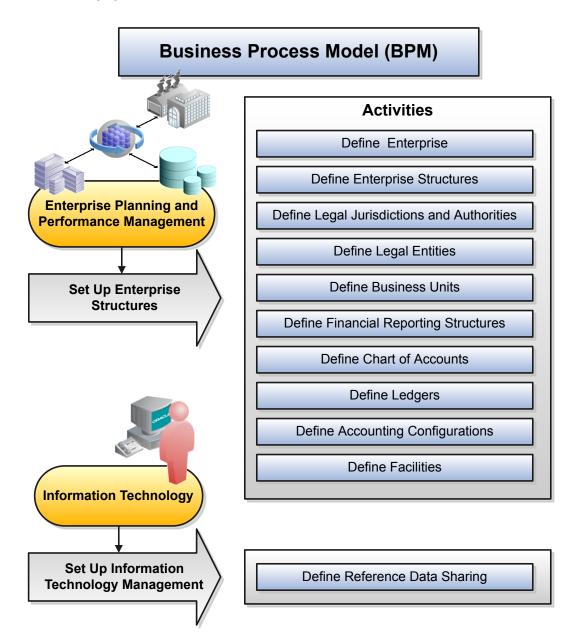
# Enterprise Structures Business Process Model: Explained

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 includes:

- Set Up Enterprise Structures business process, which consists of implementation activities that span many product families.
- Information Technology, a second Business Process Model which contains the Set Up Information Technology Management business process.
- Define Reference Data Sharing, which 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 to
  implement.



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



BPM Activities	Description
Define Enterprise	Define the enterprise to get the name of the deploying enterprise and the location of the headquarters.
Define Enterprise Structures	Define enterprise structures to represent an organization with one or more legal entities under common control. Define organizations to represent each area of business within the enterprise.



BPM Activities	Description
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 perform 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 has a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss.
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 your manufacturing and storage facilities as Inventory Organizations if Oracle Fusion tracks inventory balances there and Item Organizations if Oracle Fusion only tracks the items used in the facility but not the balances.
Define Reference Data Sharing	Define how reference data in the applications is partitioned and shared.

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

# Global Enterprise Configuration: Points to Consider

Start your global enterprise structure configuration by discussing what your organization's reporting needs are and how to represent those needs in the Oracle Fusion Applications. 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 US dollars. Your product line includes all the components to build and maintain air quality monitoring (AQM) applications 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 initial costs of these applications.

# **Analysis**

The following are elements you must 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 want 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 do you use 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 units in which to perform all these functions? How is your shared service center configured?

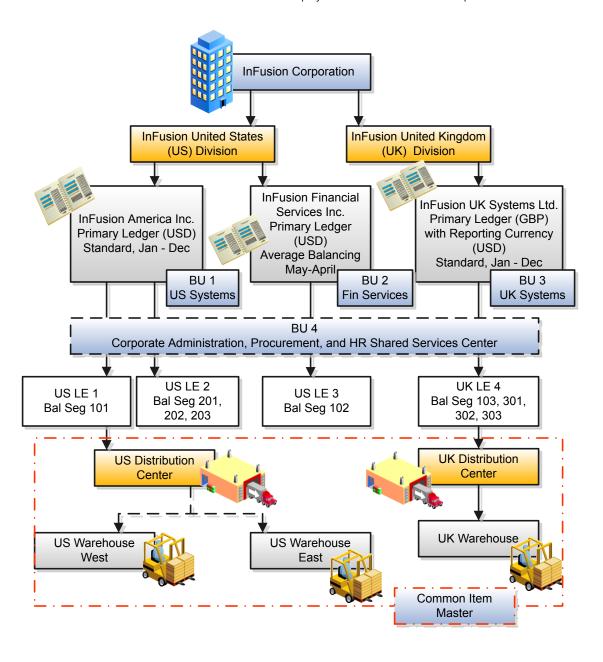
# Global Enterprise Structure Model

The following figure and table summarize the model that your committee has designed and uses numeric 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:
  - o InFusion America Inc.
  - InFusion Financial Services Inc.
  - InFusion UK Services Ltd.



- Consolidation of results for application 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





							Cost		Item	h
Doel World Entity	Entite Name	Enternales	Legal	D eV	Ladass	ВП		Dont		Inventory
Real World Entity	Entity Name	Enterprise	Entity	BOV	Ledger	ьи	Center	Dept	Oig	Org
Enterprise	InFusion Group									
Company	USLE1									
Company	USLE2		-	-						
Company	USLE3		-	7						
Company	UK LE 4		-	-						
Business Unit	U S Systems BU 1									
Business Unit	FIN Services BU 2					7				
Business Unit	UK Systems BU 3									
Division	In Fusion UK						-			
Division	InFusion US			7						
Headquarters	BU 4									
Shared Service Center	BU 4									
Department	AP Department						-			
List of Items	Common Item Master								_	
Distribution Center	U S Distribution Center									
Distribution Center	UK Distribution Center									
Warehouse	U S Warehouse West									
Warehouse	US Warehouse East									
Warehouse	UK Warehouse									
B SV = Balancing Segment Value										
BU = Business Unit										
Dept = Department										
Org = Organization										
= Mandatory Setup										
Optional Setup										

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

- The enterprise is required because it serves as an umbrella for the entire implementation. All organizations are created within an enterprise.
- Legal entities are also required. They can be optionally mapped to balancing segment values or represented by ledgers. Mapping balancing segment values to legal entities is required if you plan to use the intercompany functionality. The InFusion Corporation is a legal entity but is not discussed in this example.
- At least one ledger is required in an implementation in which you record your accounting transactions.
- Business units are also required 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 required because they track your employees.
- Optionally, add an item master organization and inventory organizations if you are tracking your inventory transactions in Oracle Fusion Applications.



Note: Some Oracle Fusion Human Capital Management and Oracle Sales Cloud implementations do not require recording accounting transactions and therefore, do not require a ledger.

# **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
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 Predefined Value Set Called Accounting Scenario	Dimension Member Name in Scenario Dimension

Even if case sensitivity is enabled in an aggregate storage outline for which duplicate member names is enabled, do not use matching dimension names with only case differences. 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 cannot be used in dimension, member, or alias names.

Character	Meaning
@	at sign
\	backslash
,	comma
-	dash, hyphen, or minus sign
	For the accounting calendar period names, you can use a hyphen or an underscore in the middle of an accounting calendar period name. For example: Jan-15 or Adj_Dec-15 can be used successfully.
=	equal sign
<	less than sign
0	parentheses
	period
+	plus sign
	single quotation mark
-	underscore
	For the accounting calendar period names, you can use a hyphen or an underscore in the middle of an accounting calendar period name. For example: Jan-15 or Adj_Dec-15 can be used successfully.
	vertical bar

# Other Restrictions

- Don't place spaces at the beginning or end of names. Essbase ignores such spaces.
- Don't use these types of words as dimension or member names:
  - o Calculation script commands, operators, and keywords.
  - Report writer commands.



- Function names and function arguments.
- o Names of other dimensions and members (unless the member is shared).
- o 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	ТО



List 1	List 2	List 3
TOLOCALRATE	TRAILMISSING	TRAILSUM
UMINUS	UPPER	VARORXMBR
XMRONLY	\$\$\$UNIVERSE\$\$\$	#MI

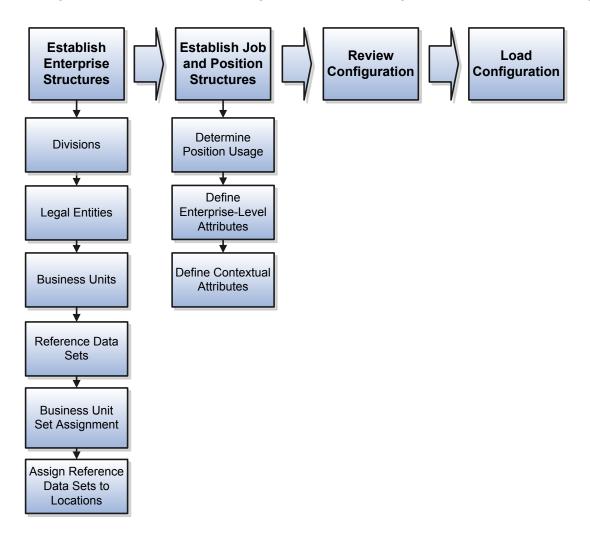
# Define Initial Configuration with the Enterprise Structures Configurator

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

### Establish Enterprise Structures

To define your enterprise structures, use the guided flow within the Establish Enterprise Structures task to enter basic information about your enterprise, such as the primary industry. 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 enough information 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, you are prompted to set up a descriptive flexfield structure for jobs, and for positions if applicable. Descriptive flexfields enable you to get more information when you create jobs and positions.

### **Review Configuration**

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

### **Load Configuration**

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

# Rolling Back an Enterprise Structure Configuration: Explained

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

### Roll Back a Configuration Manually

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

### Roll Back a Configuration Automatically

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

# Designing an Enterprise Configuration: Example

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

### Scenario

InFusion Corporation is a multinational enterprise in the high technology industry with product lines that include all the components that are required to build and maintain air quality monitoring 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 company revenue of 120 million US dollars.. Outside the US, InFusion employs 200 people and has revenue of 60 million US dollars.



### **Analysis**

InFusion requires three divisions.

- The US division covers the US locations.
- The Europe division covers UK and France.
- Saudi Arabia and the UAE are 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, 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.

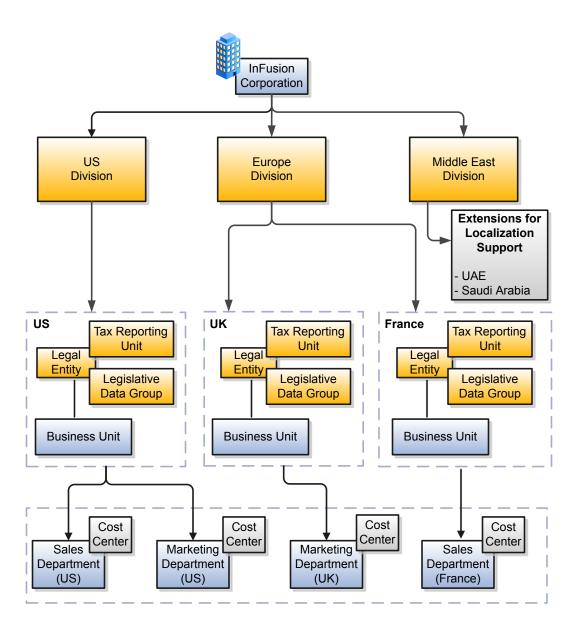
InFusion has general managers responsible for business units within each country. Those business units may share reference data. Some reference data can be defined within a reference data set that multiple business units may subscribe to. Business units are also required for financial purposes. Financial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

### Resulting Enterprise Configuration

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



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



# Divisions: 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 reflect directly the legal structure of the enterprise. The management entities and structure can include:

- Divisions and subdivisions
- Lines of business
- · Other strategic business units
- 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 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 achieving business goals including profits. 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 generally reports to a top corporate executive.

By definition a division can be represented in the chart of accounts. Companies can use 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 permit up to three balancing segments. The values of the management segment can be 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 following rights and responsibilities to:

- Own property
- Trade
- Repay debt
- 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
- Minimizing the enterprise's tax liability
- 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, 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
- 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), which 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

Use the Enterprise Structures Configurator (ESC), to 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, 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.

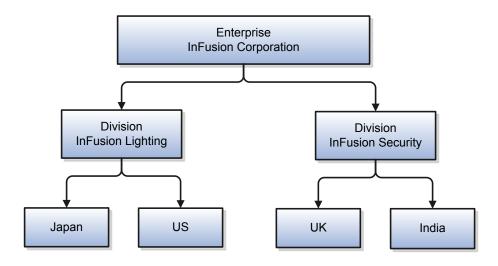
The application automatically creates the legal entities that you select, and identifies them as payroll statutory units and legal employers. For each country that you indicated that your enterprise operates in, and for each country that you created a location for, the application also automatically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



Country	Enterprise	InFusion Lighting	InFusion Security
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.

#### Related Topics

• Using Desktop Integrated Excel Workbooks: Points to Consider

# 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. These registers are part of the Oracle Fusion Trading Community Architecture.

When your legal entities are trading with each other, represent them as legal entities and 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.

Several decisions you should consider when you create legal entities.

- The importance of using legal entity on 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 Using Legal Entities on 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, a sales order creates an obligation on the legal entity that books the order to deliver the goods on the acknowledged date. The creation also creates an obligation on 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.
- 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, Business Unit ACM UK has a default legal entity of InFusion UK Ltd. When a purchase order is placed in ACM UK, the legal entity InFusion UK Ltd is legally obligated to the supplier. Oracle Fusion Procurement, Oracle Fusion Project Portfolio Management, 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 wanted, 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 one segment 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



with 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. This solution is used 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 want to isolate the assets and liabilities for that entity.

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

If your enterprise regularly spins off businesses or holds managers accountable for utilization of assets, identify the business with a balancing segment value. If you account for each legal entity in a separate ledger, no requirement exists to identify the legal entity with a balancing segment value.

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 the portion is not a separate legal entity. If you do not map legal entities sharing the same ledger to balancing segments, you cannot distinguish them using intercompany functionality or track 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 feature to create intercompany entries automatically 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 then 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.



💡 Tip: In the Oracle Fusion Supply Chain applications, you can model intercompany relationships using business units, from which legal entities are derived.

## 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 for 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 has 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 do the following:

- Assign your business units to one primary ledger. For example, if a business unit is processing payables invoices, then it must post to a particular ledger. This assignment is required for your business units with business functions that produce financial transactions.
- Use a business unit as a securing mechanism for transactions. For example, if you run your export business separately from your domestic sales business, then 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 provides the following advantages:

- Enables flexible implementation
- Provides consistent entity that controls and reports on transactions
- Shares 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 share reference data, such as payment terms and transaction types, across business units, or you can have each business unit manage its own set depending on the level at which you want 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 legal entities to ensure the uniqueness of sequencing.

In summary, use business units for:

- Management reporting
- Transaction processing
- Transactional data security
- Reference data sharing and definition

### Brief Overview of Business Unit Security

A number of Oracle Fusion Applications use business units to implement data security. You assign roles like Accounts Payable Manager to users to permit them to perform specific functions, and you assign business units for each role to users to give them access to data in those business units. For example, users which have been assigned a Payables role for a particular business unit, can perform the function of payables invoicing on the data in that business unit. Roles can be assigned to users manually using the Security Console, or automatically using provisioning rules. Business Units can be assigned to users using the Manage Data Access for Users task in Setup and Maintenance.



#### Related Topics

Reference Data Sets and Sharing Methods: Explained

## 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 one of two levels:

- Business function level, such as Sales, Consulting, Product Development, and so on.
- 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. You select the check boxes at the intersections of the two components.

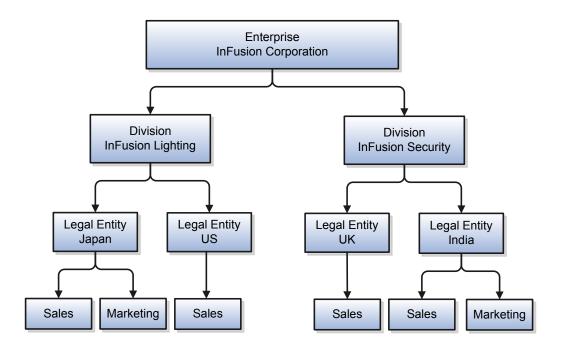
The business units listed by the application are suggestions only, and are meant to simplify the process to create business units. You aren't required to select all of the business units suggested. When you navigate to the next page in the ESC guided flow, the Manage Business Units page, you can't delete any of the business units 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 its enterprise structure. InFusion has 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><li>US</li><li>UK</li><li>Japan</li><li>India</li></ul>
Country and Division	<ul> <li>InFusion Lighting: Japan</li> <li>InFusion Lighting: US</li> <li>Infusion Security: UK</li> <li>Infusion Security: India</li> </ul>
Country and business function	<ul> <li>Sales: Japan</li> <li>Marketing: Japan</li> <li>Sales: US</li> <li>Sales: UK</li> <li>Marketing: India</li> </ul>



Business Unit Level	Suggested Business Units
	Sales: India
Division	InFusion Lighting
	InFusion Security
Division and Legal Entity	InFusion Lighting: Japan
	InFusion Lighting: US
	Infusion Security: UK
	Infusion Security: India
Division and Business Function	InFusion Lighting, Sales
	InFusion Lighting, Marketing
	<ul> <li>InFusion Security, Sales</li> </ul>
	InFusion Security, Marketing
Business Function	Sales
	Marketing
Legal Entity	Legal Entity: Japan
	Legal Entity: US
	Legal Entity: UK
	Legal Entity India
Legal Entity and Business Function	Legal Entity: Japan, Sales
	Legal Entity: Japan, Marketing
	Legal Entity: US, Sales
	Legal Entity: UK, Sales
	Legal Entity India, Marketing
	Legal Entity India, Sales
	<ul><li>Legal Entity: UK, Sales</li><li>Legal Entity India, Marketing</li></ul>

### 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.

#### Related Topics

Reference Data Sets and Sharing Methods: Explained

### 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. As a result, there is a reduction in the administrative burden and the time to implement new business units. For example, you can share sales methods, or transaction types across business units. You may also share certain 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 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 determine that certain aspects of your corporate policy can affect all business units. The remaining aspects are at the discretion of the business unit manager to implement. This allows your enterprise to balance autonomy and control for each business unit. For example, your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level. In such a case, you can let managers define their own sales methods, but define payment terms centrally. In this example:

- Each business unit has its own reference data set for sales methods.
- One central reference data set for payment terms is 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 set up 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

Variations exist 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. This method is 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. This method is 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. You need not explicitly
  assign 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 the following:
  - Transaction types from the set assigned to the business unit.
  - o 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 assign Net 15 to a set specific only to your business unit. At transaction
  entry, the list of values for payment terms consists of only 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 enables you to group set-enabled reference data such as jobs or grades to share the data across different parts of the organization. Sets also enable you to filter reference data at the transaction level so that only data assigned to certain sets is available to be selected. 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 assigned to the Common Set is always available, in addition to the reference data assigned to the set that corresponds to the business unit on the transaction.

Other types of reference data can be specific to certain business units, so you can 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.

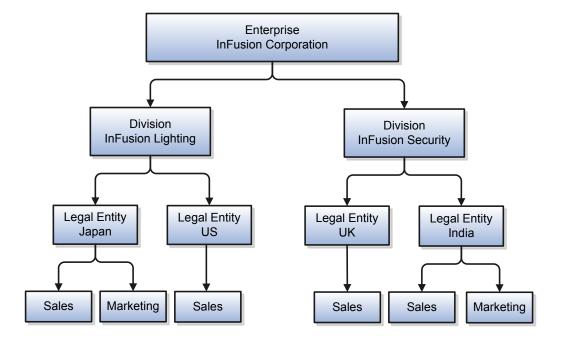
### Business Unit Set Assignment

When you assign reference data sets to business units, you assign a default reference data set to use 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 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. They 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

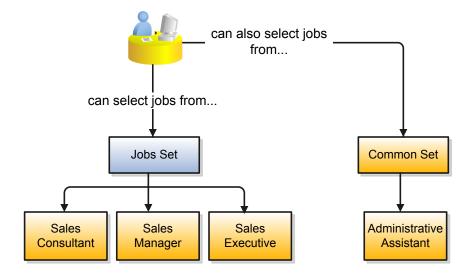
When setting up grades, departments, and locations for the business units, InFusion assigns the data to the default set for each business unit. When setting up jobs, they assign the Jobs set and assign the Common Set to any jobs that may be used throughout the entire organization.



When using grades, departments, and locations at the transaction level, users can select data from the set that corresponds to the business unit they enter on the transaction, and any data assigned to the Common Set. For example, for transactions for the Marketing\_Japan business unit, grades, locations, and departments from the Mktg\_Japan\_Set is available to select, as well as from the Common Set.

When using jobs at the transaction level, users can select jobs from the Jobs set and from the Common Set when they enter a sales business unit on the transaction. For example, when a manager hires an employee for the Sales\_India business unit, the list of jobs is filtered to show jobs from the Jobs and Common sets.

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 setenabled data at the transaction level, the list of values includes data in the:

- Common set
- 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 includes grade in the:

- Common set
- 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.

#### Note the following:

- The key to using jobs or positions depends on 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. For example, if you hire someone into a new role and want to transfer the position to another department.

During implementation, one of the earliest decisions 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

The following table outlines information about Primary industries and how they set up their workforce.

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



Primary Industry	Workforce Setup
Finance and Insurance	Jobs
riidice did iisuldice	Jous
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 employee turnover.

Industry	You always replace employees by rehiring to same role	You replace the headcount, but the manager can use the headcount in a different job	You 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



#### Related Topics

Grades and Grade Rates: How They Work with Jobs, Positions, Assignments, Compensation, and Payroll

### Positions: Examples

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

### Retail Industry

ABC Corporation has high turnovers. It loses approximately 5% of its cashiers monthly. The job of the cashier includes three positions: front line cashier, service desk cashier, and layaway cashier. Each job is cross-trained to take over another cashier's 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 Corporation must replace each cashier lost to turnover. Since turnover is high in retail it's better for this industry to use positions.

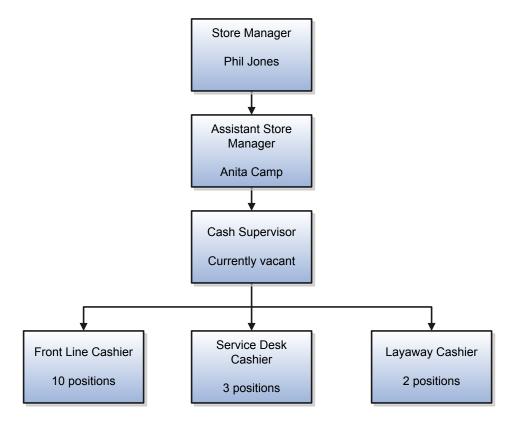
#### Note the following:

- An automatic vacancy is created when an employee terminates employment.
- The position exists even when there are no holders. Having the position continue to exist 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 the position is empty.
- You don't have to reassign these employees to another manager or supervisor. The replacement manager is assigned to the existing position.

Also, an added advantage to using Positions is when you hire somebody new, many of the attributes are inherited from the position. This speeds up the hiring process.



This figure illustrates the retail position setup.



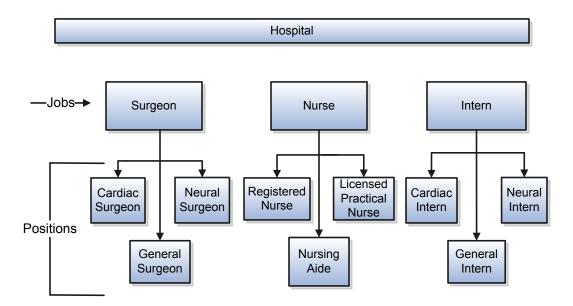
### Health Care Industry

Health care is an industry that must 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.

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



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

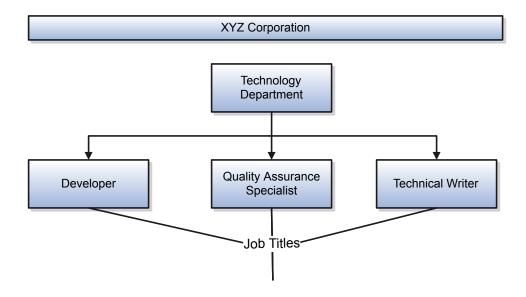
For example, XYZ Corporation has a director over the departments for developers, quality assurance, and technical writers.

- 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

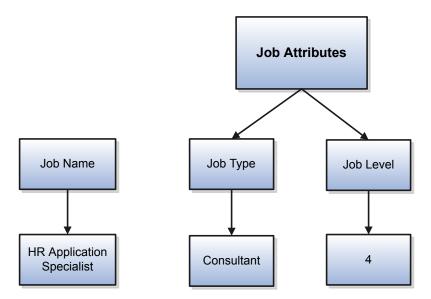
Job and position structures identify the descriptive flexfield structure that enables you to specify additional attributes that you want to capture when you define jobs and positions. Job and position attributes provide further detail to make jobs and positions more specific. You also use attributes 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 to identify additional details about the job, such as the nature of the work that is performed or the relative skill level required. 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.

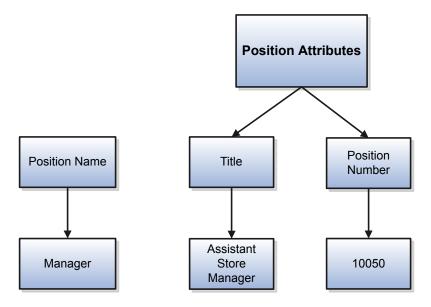


### **Enterprise-Level Position Attributes**

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. A position belongs 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, consider whether 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. 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 can't set up multiple configurations and compare different scenarios. Using the Enterprise Structures Configurator is the recommended process for setting up your enterprise structures.



### 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 these levels:

- Ultimate holding company as the top level
- Divisions and country holding companies as the second level
- 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
- 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 common data is shared or partitioned across business entities to avoid duplication and reduce maintenance effort. 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.

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 can't 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 different business units setting up and using the same grades, XYZ Corporation decides to create a set called Grades, which contains the grades. All business units in the organization have the Grades set so that the grades can be shared and used.

Note: For specific information about configuring reference data sharing for a particular object or product, refer to the relevant product documentation.

#### Related Topics

Reference Data Sets: Explained



- Reference Data Sets and Sharing Methods: Explained
- Assigning Reference Data Sets to Reference Objects: Points to Consider

### Reference Data Sets: Explained

Reference data sets are logical groups of reference data that various transactional entities can use depending on the business context. You can get started using either the common reference data set or the enterprise set depending on your implementation requirement. You can also create and maintain custom reference data sets, while continuing to use the common reference data set.

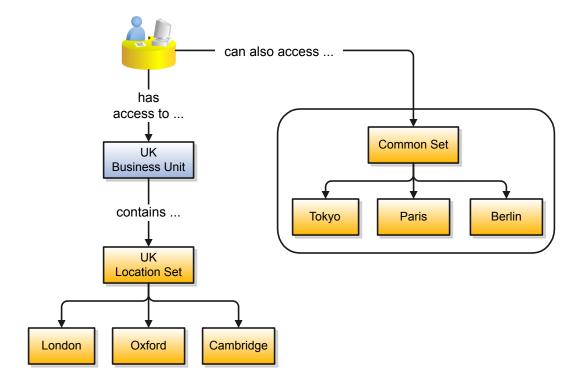
Consider the following scenario. Your enterprise can decide that only some aspects of corporate policy should affect all business units. The remaining aspects are at the discretion of the business unit manager to implement. This enables your enterprise to balance autonomy and control for each business unit. For example, your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level. Then, you can let managers define their own sales methods, but define payment terms centrally. As a result, each business unit has its own reference data set for sales methods and one central reference data set for payment terms assigned to all business units.

### Partitioning

Partitioning reference data and creating data sets provide you the flexibility to handle the reference data to fulfill your business requirements. You can share modular information and data processing options among business units with ease. You can create separate sets and subsets for each business unit. Alternatively, you can create common sets or subsets to enable sharing reference data between several business units, without duplicating the reference data.



The following figure illustrates the reference data sharing method (assignment to one set only, with common values). 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.



#### Related Topics

- Reference Data Sets and Sharing Methods: Explained
- Defining Default Reference Data Sets: Points to Consider
- Assigning Reference Data Sets to Reference Objects: Points to Consider

### 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. As a result, there is a reduction in the administrative burden and the time to implement new business units. For example, you can share sales methods, or transaction types across business units. You may also share certain 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 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 determine that certain aspects of your corporate policy can affect all business units. The remaining aspects are at the discretion of the business unit manager to implement. This allows your enterprise to balance autonomy and control for each business unit. For example, your enterprise holds business unit managers accountable for their profit and loss, but manages working capital requirements at a corporate level. In such a case, you can let managers define their own sales methods, but define payment terms centrally. In this example:

- Each business unit has its own reference data set for sales methods.
- One central reference data set for payment terms is 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 set up 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

Variations exist 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. This method is 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. This method is 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. You need not explicitly
  assign 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 the following:
  - Transaction types from the set assigned to the business unit.
  - 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 assign Net 15 to a set specific only to your business unit. At transaction
  entry, the list of values for payment terms consists of only 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 using 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 the 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 using a business context setting called the determinant type. A determinant type is the point of reference used in the data assignment process. The following table lists the determinant types used in the reference data assignment.

Туре	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 (also called determinant value) is a value that corresponds to the selected determinant type. The determinant is one of the criteria for selecting the appropriate reference data set.

### Reference Groups

A transactional entity may have multiple reference entities (generally considered to be setup data). However, all reference entities are treated alike because of similarity in implementing business policies and legal rules. Such reference entities in your application are grouped into logical units called reference groups. For example, all tables and views that define Sales Order Type details might be a part of the same reference group. Reference groups are predefined in the reference groups table.

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

Some products, such as items and supplier sites, required special logic for reference data sharing and have implemented their own domain-specific ways for sharing data.

#### Items

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

### Suppliers Sites

You can approve particular suppliers to supply specified commodities and authorize your business units to buy from those suppliers when the need arises. For example, you might be a household cleaning products manufacturer and need dyes, plastics, and perfumes to make your products. You purchase from a central supplier 70% of your perfume supplies with an



additional supplier, in reserve, from whom you purchase the remaining 30%. At the same time, each of your business units purchases plastics and dyes from the same supplier, but from different local supplier sites to save transportation costs.

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

- Supplier qualification data, such as approved supplier lists
- · Catalog content, such as agreements, smart forms, public shopping lists, and content zones
- Procurement configuration data

### FAQs for Define Reference Data Sharing

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

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

Application Name	Reference Data Object	Method of Sharing
Trading Community Model	Customer Account Relationship	Assignment to one set only, no common values allowed
Trading Community Model	Customer Account Site	Assignment to one set only, no common values allowed
Trading Community Model	Salesperson	Assignment to one set only, no common values allowed
Opportunity Management	Sales Method Group	Assignment to one set only, with common values
Work Management	Assessment Templates	Assignment to one set only, with common values
Enterprise Contracts	Contract Types	Assignment to one set only, with common values
Sales	Sales Method	Assignment to one set only, with common values
Common Components	Activity Templates	Assignment to one set only, with common values
Payables	Payment Terms	Assignment to multiple sets, no common values allowed



Application Name	Reference Data Object	Method of Sharing
Receivables	Accounting Rules	Assignment to one set only, with common values
Receivables	Aging Buckets	Assignment to one set only, with common values
Receivables	Auto Cash Rules	Assignment to one set only, with common values
Receivables	Collectors	Assignment to one set only, with common values
Receivables	Lockbox	Assignment to one set only, with common values
Receivables	Memo Lines	Assignment to one set only, with common values
Receivables	Payment Terms	Assignment to one set only, with common values
Receivables	Remit To Address	Assignment to one set only, with common values
Receivables	Revenue Contingencies	Assignment to one set only, with common values
Receivables	Transaction Source	Assignment to one set only, with common values
Receivables	Transaction Type	Assignment to one set only, with common values
advanced Collections	Collections Setups	Assignment to one set only, with common values
advanced Collections	Dunning Plans	Assignment to one set only, with common values
ax	Tax Classification Codes	Assignment to multiple sets, no common values allowed
Human Resources	Departments	Assignment to one set only, with common values
luman Resources	Jobs	Assignment to one set only, with common values



Application Name	Reference Data Object	Method of Sharing
Human Resources	Locations	Assignment to one set only, with common values
Human Resources	Grades	Assignment to one set only, with common values
Project Billing	Project and Contract Billing	Assignment to multiple sets, no common values allowed
Project Foundation	Project Accounting Definition	Assignment to one set only, no common values allowed
Project Foundation	Project Rates	Assignment to one set only, with common values
Order Management	Hold Codes	Assignment to one set only, with common values
Order Management	Orchestration Process	Assignment to one set only, with common values

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

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

Application Name	Reference Data Object	Method of Sharing
Assets	Bonus Rules	Assignment to one set only, no common values allowed
Assets	Depreciation Ceilings	Assignment to one set only, no common values allowed
Assets	Depreciation Methods	Assignment to one set only, with common values
Assets	Asset Descriptions	Assignment to one set only, no common values allowed
Assets	Property Types	Assignment to one set only, with common values
Assets	Prorate Conventions	Assignment to one set only, no common values allowed
Assets	Asset Queue Names	Assignment to one set only, with common values
		valdoo



Application Name	Reference Data Object	Method of Sharing
Assets	Retirement Types	Assignment to one set only, with common values
Assets	Unplanned Types	Assignment to one set only, with common values

### What reference data objects can be shared across cost organizations?

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

Application Name	Reference Data Object	Method of Sharing
Cost Management	Cost Structure	Assignment to one set only, no common values allowed

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

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

Application Name	Reference Data Object	Method of Sharing
Project Foundation	Project Definition	Assignment to multiple sets, no common values allowed
Project Foundation	Project Transaction Types	Assignment to multiple sets, no common values allowed

### Define Enterprise: Manage Enterprise HCM Information

### Enterprise: Explained

An enterprise is a collection 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. 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 create and manage locations using the Manage Locations task in the Workforce Structures work area.

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 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**

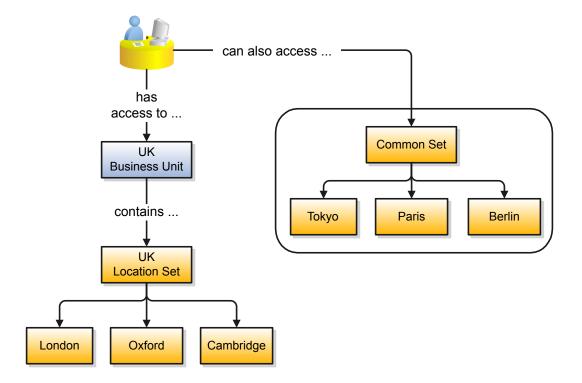
When you create a location, you must associate it with a set. Only those users who have access to the set's business unit can access the location set and other associated workforce structure sets, such as those that contain departments and jobs.

#### Note the following:

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



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



### Uploading Locations Using a Spreadsheet

If you have a list of locations already defined for your enterprise, you can upload them from a spreadsheet.

To use this option:

- Download a spreadsheet template
- Add your location information to the spreadsheet
- Upload directly to your enterprise configuration

You can upload the spreadsheet multiple times to accommodate revisions.

#### Related Topics

Uploading Workforce Structures Using a Spreadsheet: Explained

### 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 you created for the geographic node start to apply for the location and may impact the availability of worker assignments at that location. You manage locations using the Manage Locations task in the Workforce Structures work area.

The geographical hierarchy nodes available for selection on the Locations page display from a predefined geographic hierarchy.

#### Related Topics

Worker Availability: How It Is Determined

### 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 Oracle Fusion Global Human Resources, go to the Manage Locations page. Use the Manage Locations task in the Workforce Structures work area.

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

### Define Geographies

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

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

### Geography Structure

Firstly, you need to create a geography structure for each country to define which geography types are part of the country structure, and how the geography types are hierarchically related within the country structure. For example, you can create geography types called State, City, and Postal Code. Then you can rank the State geography type as the highest level within the country, the City as the second level, and the Postal Code as the lowest level within the country structure. Geography



structure can be defined using the **Manage Geographies** task, or can be imported using tasks in the **Define Geographies** activity.

### Geography Hierarchy

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

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

### Geography Validation

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

### Geography Structures: Explained

This topic describes geography structures and the tasks you can perform using geography structures.

A geography structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States is as follows:

Level	Geography Type
1	State
2	County
3	City
4	Postal Code

You can use the geography structure to relate geography types for a country and define geography types for a country.

### Relate Geography Types for a Country

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 geography type Country is implicitly at the top of the geography structure with the level 1. The subsequent geography types that you add after country are numbered in sequence.

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. To quickly create country structure, you can copy a structure from another country and modify the geography types for the country.



### Define Geography Types for a Country

You can use any of the master reference geography types to create your geography structure. 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: You cannot delete geography types that have associated geography data. You can only delete the lowest level geography type of the country structure.

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

### Geography Hierarchy: Explained

This topic describes geography hierarchy and various aspects of geography hierarchy.

Geography hierarchy is a data model that creates conceptual parent-child relationships between geographies. The top level of the geography hierarchy is country, which is the parent, and the hierarchy contains several child geographies. The following table shows sample parent-child relationships in a geography.

California	Parent of San Mateo county
San Mateo County	Parent of Redwood City
Redwood City	Parent of 94065
94065	Child

When you enter just 94065, the application determines that the postal code is in California and the corresponding city is Redwood City.

The application uses geography hierarchy information to facilitate business processes that rely on geography information, such as, tax calculation, order sourcing rules, and sales territory definition. The geography hierarchy information is centrally located and shared among other application offerings.

The geography hierarchy includes:

- Geography: Geography is a physical space with boundaries that is a defined instance of a geography type, such as country, state, province or city. For example, San Jose is a geography of the City geography type.
- Geography type: Geography types are divisional grouping of user defined geographies, for example, Continent, Country Regions, and Tax Regions.
- Geography usage: Geography usage indicates how a geography type or geography is used in the application.
- Master reference geography hierarchy: The geography hierarchy data is considered the single source of reference for all geography related data such as geography types and geographies.

The geography usage for the entire hierarchy is the master reference, and defined geography types and geographies are the master reference geography types and geographies. For example, you can create geography types called State, City, and Postal Code. Then, you can rank the State as the highest level, City as the second level, and Postal Code as the lowest level within the country structure.



• User defined zones: User defined zones are a collection of geographical data, created from master reference data for a specific purpose. For example, while the territory zones are collections of master reference geographies ordered with a hierarchy, the tax and shipping zones are 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. If the setup for master geography data is incomplete, then the geography data is either not imported or created. As a result, the list of values for the address attribute does not list any geography data.

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.

The Geography Dimension value in territories is derived from sell-to addresses of sales accounts. To use geography dimensions in territories, you must validate the geography elements in the addresses, such as state, city, and postal code. You can validate the address by enabling geography validation for each country using the Manage Geographies task. Perform the following in the Manage Geographies task:

- Enable at least one level in the geography hierarchy for geography validation.
- Enable geography validation for all geography levels that you intend to use for territory definition for each country.
- If needed, 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 set geography validation control to Error in the Manage Geography Validation page. This ensures that users can only use valid geography elements in addresses.

Note: If you have already created addresses before setting up geography validation for a country, you must enabling geography validation and then execute the Run Maintain Geography Name Referencing task for that country. This validates all your geography elements.



## 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:	
	1. County	
	2. Post Town	
What is the geography hierarchy?	Create the following hierarchy:	
	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 <b>Tax Validation</b> for the geography types.	

### 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**.
- 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 create the geography hierarchy for United Kingdom, 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.

- On the Manage Geographies page, enter GB in the Code field. Click Search.
- 2. On the Manage Geographies page, click **Hierarchy Defined**.
- 3. In the Geography Hierarchy section, click United Kingdom to highlight the table row, and click Create.



- 4. In the Create County page, Primary and Alternate Names section, enter Berkshire in the Name field.
- 5. Click Save and Close.
- 6. In the Geography Hierarchy section, click Berkshire to highlight the table row, and click Create.
- 7. In the Create Post Town page, Primary and Alternate Names section, enter Reading in the Name field.
- 8. Click Save and Close.

### Defining the Geography Validations

To specify the geography validations for the geography types you added to 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. In the 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. Select the **Enable List of Values** and **Geography Validation** options.
- 5. For the Post Town geography type, click the City list item in the Map to Attribute field.
- **6.** Select the **Geography Validation** option.
- 7. In the Geography Validation Control section, select Error in the Geography Validation Level for Country list.
- 8. Click Save and Close.

### Geocoding: Explained

This topic explains geocoding and how to enable this option in the application.

Geocoding is the process of finding latitude and longitude coordinates from geographic data such as street addresses or zip codes. Once these coordinates are available, you can use the spatial services feature to identify points of interest, such as customer and contact addresses, in the vicinity. The application integrates the Geocoding feature with eLocation (http://elocation.oracle.com/maps\_oracle\_dot\_com\_main.html), which is a Geocoding service provided by Oracle.

By default, the Geocoding option is turned off in the application. You can enable the Geocoding option in the **Setup and Maintenance, Manage Geographies** page.

If the Geocoding feature is enabled, the feature can be scheduled to run at regular time intervals. This ensures that newly created or updated locations are picked up and geocoded whenever you create or update an address using the user interface, web services, bulk import, or file-based import.

#### Related Topics

• What are Spatial Services?

### Setting Up Geocoding: Procedure

This procedure lists the steps to set up geocoding in Oracle applications.

Geocoding is a process that determines the latitude and longitude coordinates for a location. By default, geocoding is turned off in the application. You can use geocoding to display customers in the vicinity of a mobile address.

### Enabling Geocoding for a Country

To enable geocoding for a country, complete these steps:

1. From the Setup and Maintenance work area, search for Manage Geographies and click **Go to Task**.



- 2. Search the country for which you want to enable geocoding. You can either search by the country name or country code.
- 3. Click **Search**. The search results for the matching country names are displayed.
- 4. Select the country for which you want to enable the geocoding option.
- 5. Select **Geocoding Defined** for the country.

### Populating Location Latitude and Longitude Information

Once geocoding is enabled, you can schedule this feature to run at regular time intervals so that newly created or updated locations are picked up and geocoded. To schedule the geocoding feature to run at regular intervals, complete these steps:

- 1. Navigate to the Scheduled Processes work area, and click **Schedule New Process**.
- 2. Click the Name drop-down and search for Populate Location Latitude and Longitude Information, and then click OK.
- 3. Enter the parameters such as Start Date and End Date, and click **Submit**.

### 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:

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

### Nokia Geography Reference Data

Oracle Sales Cloud includes third-party (Nokia) master geography data for multiple countries that can be easily imported. You can import Oracle-licensed Nokia data from Navteq, for those countries where the data is available, such as the U.S. You can import Nokia Geography data using the **Manage Geographies** task. Search for the country, and select **Import Nokia Data** from the **Actions** menu. If the licensed Navteq data is not available for a particular country, then the **Import Nokia Data** action is disabled.

### 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.

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_T	HZ_GEOGRAPHIES	Geography
		HZ_ GEOGRAPHY_ IDENTIFIERS	
		HZ_ GEOGRAPHY_ TYPES_B	
		HZ_ HIERARCHY_ NODES	

#### Related Topics

• File-Based Import Processing: How it Works

### Nokia Geography Reference Data: Explained

Oracle Sales Cloud includes third-party Nokia master geography data that is available for import for the following countries.

	Country Name	Country Code
1.	Andorra	AD
2.	Argentina	AR
3.	Austria	AT
4.	Belgium	BE
5.	Brazil	BR
6.	Bulgaria	BG
7.	Canada	CA
8.	Cayman Island	КУ
9.	Chile	CL
10.	Croatia	HR
11.	Czech Republic	CZ



	Country Name	Country Code
12.	Denmark	DK
13.	Dominican Republic	DO
14.	Estonia	EE
15.	Finland	FI
16.	France	FR
17.	Germany	DE
18.	Greece	GR
19.	Guadeloupe	GP
20.	Hungary	HU
21.	Iceland	IS
22.	India	IN
23.	Indonesia	ID
24.	Ireland	IE
25.	Isle of Man	IM
26.	Israel	IL
27.	Italy	ιπ
28.	Jamaica	JM
29.	Latvia	LV
30.	Liechtenstein	Ц
31.	Lithuania	LT
32.	Luxembourg	LU
33.	Malaysia	MY



	Country Name	Country Code
34.	Malta	MT
35.	Martinique	MQ
36.	Mexico	MX
37.	Netherlands	NL
38.	New Zealand	NZ
39.	Norway	NO
40.	Peru	PE
41.	Poland	PL
42.	Portugal	PT
43.	Puerto Rico	PR
44.	Reunion Island	RE
45.	Romania	RO
46.	Russian Federation (Russia)	RU
47.	San Marino	SM
48.	Slovakia	SK
49.	Slovenia	SI
50.	South Africa	ZA
51.	Spain	ES
52.	Swaziland	SZ
53.	Sweden	SE
54.	Switzerland	CH
55.	Taiwan	TW



	Country Name	Country Code
56.	Turkey	TR
57.	United Kingdom	GB
58.	United States	US
59.	Uruguay	UY
60.	Holy See (Vatican City State)	VA

# Replacing Existing Master Geography Data with Revised Nokia Geography Data: Procedure

You must import and set up reference geography data for the countries where you do business. Using the Nokia geography reference data, you no longer have to source geography data from a third party. You can import Oracle-licensed Nokia data from NAVTEQ, including the country structure and hierarchy information, either to create a new geography setup or replace your existing geography data.

This topic describes the steps to replace your existing master geography data with the revised Nokia geography data.

## Creating an Export File of All Territories

You must export all territories before deleting the master geography data because removing the master geography data invalidates the territory definitions that are based on the Geography dimension. You can either export the definitions of all territories to a file or make manual corrections. If there are a large number of territories, export the territories definition to a file that can be used during the territories import process. However, if there are very few affected territories, then you can choose to either export the territories definition to a file or make corrections manually.

This procedure is applicable only if there are territories defined using the Geography dimension.

Perform the following steps to create an export file of all territories.

- 1. From the Territories and Quotas work area, click View Active Territories in the Tasks pane.
- 2. In the View Active Territories page, select the top territory.
- 3. Click the Actions drop-down list, and select Export, and then Export Selected Territory Hierarchy.
- 4. In the Warning dialog box, click **OK**.
- 5. Click the **Actions** drop-down list and select **Export**, and then **View Export** Status.
- **6.** Review the status of the export job and verify if it has completed successfully.
- 7. In the Exported Data File column, click the .zip file against your export job, and click **Save**. All the territories are exported to a compressed file on your system.
- 8. Click OK.
- 9. Click **Done** in the View Active Territories page.



#### Deleting the Territory Geography Data

A territory definition has references to the territory geography data and master geography data. Since territory geography data is based on the master geography data, you must delete the territory geography data prior to deleting the master geography data. When you delete the territory geography data, all territories that are defined using geography dimension become invalid.

This procedure is applicable only if territory geographies are defined.

Perform the following steps to delete the territory geography data.

- 1. From the Setup and Maintenance work area, search for Manage Territory Geographies and click Go to Task.
- 2. In the Manage Territory Geographies page, click View All Hierarchies.
- 3. Select the top node for the country for which you want to replace the master geography data and click the **Delete** icon.
- 4. In the Warning dialog box, click **OK**.
- 5. In the Confirmation dialog box, click **OK**. The parent node of the territory geography data and its children are deleted.
- 6. Repeat steps 3 to 5 to delete all top nodes in the territory geography data.
- 7. Click Save and Close.

Although the territory geography data is deleted, the territory definitions may appear to remain valid. This is because the Territory Management application retains a copy of the dimension members referenced in the territory definitions. This copy is updated when you trigger the **Load and Activate** process from the **Enable Dimensions and Metrics** task.

#### Deleting the Master Geography Data

To delete the master geography data for a country, you must create a support request with proper justification. Note that when the master geography data is deleted, the geography and its children are deleted and all the related territory, tax, and shipping zone references become invalid. So you must take backup of these before deleting the master geography data.

## Importing Nokia Geography Reference Data

Use this procedure to import Nokia geography reference data licensed by Oracle. If the country data you want to import is not available, then the Import Nokia Data action is disabled.

The geography data is provided by Nokia and is third-party content. As per Oracle policy, this software and documentation may provide access to or information about content and services from third parties. Oracle and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content and services. Oracle and its affiliates are not responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Perform the following steps to import Nokia geography reference data. Currently, the revised Nokia geography reference data is available only for US at this time.

- 1. From the Setup and Maintenance work area, search for Manage Geographies, and click Go to Task.
- 2. In the Manage Geographies page, enter either the country name or the two-letter ISO code (for example, US), and click **Search**.
- **3.** Select the country in the search results.
- 4. Click the Actions drop-down list, and select Import Nokia Data.
- 5. In the Warning dialog box, click **OK**.
- 6. In the Confirmation dialog box, click **OK**.

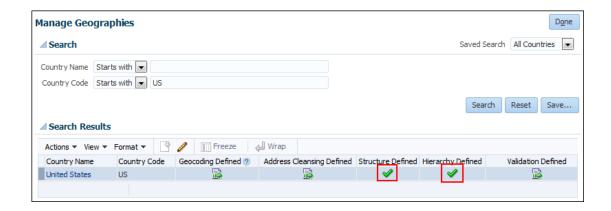
The import of larger countries may require several hours to complete.

You can track the progress of the import process by selecting **Scheduled Processes** from the Navigator menu.



Note: To access the Scheduled Processes work area, you must be signed in as a user with the Employee abstract role. The initial user does not have this role assigned, but the other users you created do.

After the import is complete, you can search for the country again in the Manage Geographies page. Check marks now appear in the **Structure Defined** and **Hierarchy Defined** columns indicating the import completed successfully.



Next, click the **Validation Defined** icon to define the validations, enable List of Values, and choose address style format for a country as set up before. For more information, see the "Geography Validation: Explained" topic.

The **Geocoding Defined** and **Address Cleansing Defined** columns are used for additional features which you must license from Oracle and set up separately.

- Geocoding makes it possible to display customers in the vicinity of a mobile address. You set up **Geocoding** Enabled for those countries where you are using Around Me functionality in Sales Cloud Mobile.
- Cleansing makes it possible to validate addresses down to the street level.

## Running the Geography Name Referencing Process

The Geography Name Referencing (GNR) process validates address elements in location tables, such as HZ\_LOCATIONS, against the master geography data.

Perform the following steps to run the GNR process.

- 1. Navigate to the Scheduled Processes work area, and click Schedule New Process.
- Click the Name drop-down list and search for Validate Geographies Against Master Geographies, and then click OK.
- 3. Click **OK** in the Schedule New Process dialog box.
- 4. In the Process Details dialog box, enter the following details:
  - Location Table Name: HZ\_LOCATIONS
  - Run Type: ALL
  - Usage Code: GEOGRAPHY
- 5. Enter the country code in the Country Code field.
- 6. Click Submit.
- 7. In the Confirmation dialog box, click **OK**.
- 8. Click Close.
- 9. In the Scheduled Processes page, click the Refresh icon.



10. Verify if the status of the process has completed successfully.

## Recreating and Loading the Territory Geography Data

You can recreate the territory geography data, after the master geography data is imported, using either of the following methods:

- Import process: If you created the original territory geography data using the import process, then use the same import file to recreate the territory geography structure.
  - For more information about importing the territory geography data using the import file, see "Importing Territory Geography Hierarchies Using File-Based Data Import: Quick Start" in the Oracle Sales Cloud Understanding File-Based Data Import and Export guide.
- Manual creation process: You can manually recreate the territory geography data structures, as they existed before their deletion, using the Manage Territory Geographies task.

For more information about creating zones and adding geographies to a zone, see "Managing Territory Geographies: Worked Example" topic.

After you have recreated the territory geography data, perform the following steps to load the data.

- 1. From the Setup and Maintenance work area, search for Enable Dimensions and Metrics, and click Go to Task.
- In the Enable Dimensions and Metrics page, click the Actions drop-down list, and select Load and Activate.
  The process loads the territory geography data to make dimension members available for selection when defining territories.
- 3. In the Confirmation dialog box, click OK.
- 4. Click Done.

#### Restoring the Invalid Territory Definitions

After recreating the territory geography hierarchies and running the Load and Activate option from the **Enable Dimensions** and **Metrics** task, the geography dimensions are populated with the new geography members. The geography members in the territory appear as invalid because your territories still reference the old copies of the dimension members that were deleted. The new members are not referenced automatically by the territories. You must re-reference the territory definitions from the old geography dimension members to the new ones.

You can restore the invalid territory definitions by either importing the previously created export file or making manual corrections to the territories.

- Restoring Valid Territory Definitions Using Territories Import
  - **a.** Open the export file you saved in the "Creating an Export File of All Territories" step. The compressed file contains four CSV files.
  - b. Open TERR HEADER.CSV file.
  - c. Enter **REPLACE** in the Action column for all territories that are based on geography dimension.
  - **d.** Save the file in CSV format and compress it together with three other CSV files.
  - e. From the Territories and Quotas work area, click View Active Territories in the Tasks pane.
  - f. Click the Actions drop-down list, and select Import to Proposal, and then Import Territories.
  - g. Select the newly created compressed file and click **OK**.
  - h. Click the Actions drop-down list and select Import to Proposal, and then View Import Status.
  - i. Review the status of the export job and verify if it has completed successfully.
  - j. Click **OK**.
  - k. From the Tasks pane, click Manage Territory Proposals.



- I. In the Manage Territory Proposals page, under Current Territory Proposals table, search for the proposal with your import file name.
- m. Click the import file name to open the territory proposal.
- **n.** Click **Edit Coverage** to verify that the territory definitions are valid.
- o. Verify that there are no values listed as invalid in the Selected Dimension Members section.
- p. Click Save and Close.
- q. Click Activate. The territory proposal of your import file is activated.
- Restoring Valid Territory Definitions through Manual Corrections

Although this method is always applicable, it is most appropriate when you have to restore territory definitions for a smaller number of territories.

- a. From the Territories and Quotas work area, click Manage Territory Proposals in the Tasks pane.
- **b.** In the Manage Territory Proposals page, click the **Create** icon.
- c. In the Create Territory Proposals dialog box, enter a name and click Save and View.
- **d.** In the Territory Proposals page, add all the territories with the Geography dimension value other than the value "Any" to the proposal.
- e. Select a territory and click Edit Coverage.
- **f.** In the Edit Coverage page, select **Geography** from the Dimensions drop-down list. The invalid dimension members are displayed in the Selected Dimension Members pane.
- **g.** Expand the values in the Available Dimension Members section or search for the member that has the same name as the one marked invalid in the Selected Dimension Members pane.
- h. Select one or more new geography dimension members from Available Dimension Members pane and click **Add** icon to the Selected Dimension Members pane.
- i. Click the **Remove** icon to remove the invalid members from the Selected Dimension Members pane.
- i. Click Save and Close.
- **k.** Repeat steps 4 to 10 for all territories that were based on Geography dimension.
- I. Click **Activate**. After the activation process is complete, your territory definitions are valid again and are referencing to the new geography data.
- Running Batch Assignment Process for Opportunities
  - a. From Navigator, click Scheduled Processes.
  - **b.** In the Schedule Processes page, click **Schedule New Process**.
  - c. In the Schedule New Process dialog box, search for the Revenue Territory Based Assignment process and select it.
  - d. Click OK.
  - **e.** In the Process Details dialog box, enter **OpenOpportunitiesByCreationDate** in the View Criteria Name field. This selects all revenue lines belonging to open opportunities that were created in the last 'X' days.
  - **f.** Enter **BindOptyCreationDateFrom=** followed by the date.

For example, if BindOptyCreationDateFrom=2014-01-01, then all open opportunities which were created between 1st January 2014 till the current date, are processed.

- g. Click Submit to schedule the process.
- h. In the Confirmation dialog box, make a note of the process identifier for monitoring the process, and click **OK**.
- i. Click Close.
- j. In the Schedule Processes page, click the **Refresh** icon.
- k. Review the status of the process job and verify if it has completed successfully.



- Note: Review a small subset of the open opportunities to confirm that the territory assignment is as expected.
- Running Batch Assignment Process for Sales Accounts
  - a. Ensure that the ZCA\_SA\_AUTO\_ASSIGN\_ON\_CREATE and ZCA\_SA\_AUTO\_ASSIGN\_ON\_UPDATE profile options are set to Yes in the Manage Customer Center Profile Options task.
  - b. From Navigator, click Customers.
  - **c.** In the Customers page, click **Create Account**.
  - In the Create Account page, enter a name and address of the sales account, and select the Address is sell to check box.
  - e. Click Save and Close.
  - f. From Navigator, click Customers.
  - g. In the Search pane, search for the name of the sales account you created and select it.
  - h. Under Customer Information, select Sales Account Team. The details of the sales account and territories associated with the sales account are displayed.

This indicates that the sales account was created successfully and the batch assignment was run automatically to assign the matching territories to the sales account.

To run the batch assignment process manually from the Scheduled Processes page, perform the following steps.

- a. From Navigator, click Scheduled Processes.
- b. In the Schedule Processes page, click **Schedule New Process**.
- c. In the Schedule New Process dialog box, search for the Request Sales Account Assignments process and select it.
- d. Click OK.
- Enter SalesAccount\_Work\_Object in the Work Object Code field and
   SalesAccountTerritory\_Candidate\_Object in the Candidate Object Code field.
- **f.** Select **Territory** in the Assignment Mode list.
- g. Enter AllSalesAccountsVC in the View Criteria Name field. This selects all sales accounts.
- h. Click **Submit** to schedule the process.
- i. In the Confirmation dialog box, make a note of the process identifier for monitoring the process, and click **OK**.
- j. Click Close.
- **k.** In the Schedule Processes page, click the **Refresh** icon.
- I. Review the status of the process job and verify if it has completed successfully.
  - Note: Review a small subset of the accounts to confirm that the territory assignment is as expected.

#### Related Topics

Managing Territory Geographies: Worked Example



## Creating Countries: Procedure

This procedure lists the steps to create countries in Oracle Sales Cloud.

In Oracle Sales Cloud, countries are pre-seeded. If you are unable to find a specific country in the Manage Geographies page, then you can add it to the application.

Note: Oracle Sales Cloud provides support for Nokia geography data for 60 countries. For more information on the list of countries, see the Nokia Geography Reference Data: Explained topic. For countries where Nokia geography data is not available, you can purchase the geography data from a third-party data provider and load it into the application using File-Based Data Import. For more information, see the Importing Geographies chapter in the Oracle Sales Cloud Understanding File-Based Data Import and Export guide. If countries are not available in the application, then use the procedure outlined in this topic to create them.

Perform the following steps to create a new country.

- 1. From the Setup and Maintenance work area, search for Manage Territories and click **Go to Task**.
- 2. Click the New icon.
- 3. Enter the following details:
  - Territory Code: Enter a unique code for the territory.
  - Territory Name: Enter a unique name for the territory.
  - Description: Enter a description for the territory.
- 4. Click Save and Close.
  - Note: After you have added a new country in the application, if you want to import the geography data for that country, then you must perform Step 5 to 10.
- 5. From the Setup and Maintenance work area, search for Manage Geographies and click Go to Task.
- 6. In the Manage Geographies page, enter either the country name or the two-letter ISO code for the country you just added, and click **Search**.
- 7. Select the country in the search results.
- 8. Click the **Actions** drop-down, and select **Create Country**.
- 9. In the Create Country dialog box, select the name of the country and click **Save**.
- 10. Click Done.

## Importing Geographies Using File-Based Import: Explained

This topic explains how to prepare and import geography data from an external data source using the File-Based Data Import feature. A geography is any region with a boundary around it, regardless of its size. It might be a state, a country, a city, a country, or a ward. You must create or import geographies before you can associate them with custom zones and addresses.



Note: The application ships with third-party (Nokia) master geography data for multiple countries that can be easily imported. You can import Oracle-licensed Nokia data from Navteq, for those countries where the data is available, such as the U.S. You can import Nokia Geography data using the Manage Geographies task. Search for the country, and select Import Nokia Data from the Actions menu. If the licensed Navteq data is not available for a particular country, then the Import Nokia Data action is disabled. For more information, see Replacing Existing Master Geography Data with Revised Nokia Geography Data: Procedure. If Nokia geography data is not available for a country, then use the information in this chapter to import it using File-Based Data Import.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography data compared to how Oracle applications represent the same data?
- Do you have to configure values in the application 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 the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a geography.

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

## Import Objects for the Geography

To facilitate importing geographies, the application incorporates the structure of the geography into import objects. The import object for the geography is ImpGeography.

## Comparing Business Object Data

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

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each import object attribute. The validation information includes the navigation path to the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, 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
ImpGeography	Geography Import Objects: How They Work Together





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

#### Extensible Attributes

The application doesn't support extensible attributes for geographies. You can import only data for geography object that already exist by default in the application.

#### 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 a file-based import. The file-based import process reads the data 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. When you're 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 applications provide File-Based Import activity reports, which you can use 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.

#### Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview

## 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 objects for the Geography import object
- Target import object attributes
- · Reference guide files for target import object attributes

## Geography Target Import Objects

You can use the Geography import object 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 the application, you must understand how the target objects are related and what attributes are included in each target object.

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 (

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

#### Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs\_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A 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 prerequisite 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 Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities 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 Application Composer extensibility features for geography.

#### Reference Files for Target Import Object Attributes

To access the reference guide files for the geography's target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs\_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. 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 files that are available by target import object.

Target Import Object	Description	Attribute Reference Guide File Names
ImpGeography	Contains information that captures a country's geography hierarchy details, such as geography type, geography code, etc.	HZ_IMP_GEOGRAPHIES_T_Reference

#### Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview

# 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 example, you have a source file containing geography data that you want to import into the application so that the geography data can be used for real time address validation and tax purposes.



The following table summarizes the key decisions that you must make in 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 the application?	All, except for the RecordTypeCode field
When do you want to process the import?	Immediately

## Summary of the Tasks

You perform the following steps to create an import activity and activate the import:

- 1. Determining what information is in the source file.
- 2. Creating and scheduling the import activity.
- 3. Monitoring the import results.

## Prerequisites for Importing Additional Geography Data After Your Initial Import

- 1. Ensure that the combination of the Source ID and Parent Source ID values is unique for each row of data within a single import. However, your source data files don't 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, then the changed IDs will not affect the import.
- 2. Ensure that all 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 must 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, then your geography import creates two state records (CA and California) in the application data, with the US as the country parent.

## Determining What Information is in the Source File

- 1. The source geography data files must include a unique Source ID value for each row of data and Parent Source ID value for the parent of that row of data. The Source or Parent Source IDs should not be longer than 18 characters.
- 2. You can structure your geography source data as follows:

Geography Level	Name	Source ID	Parent Source ID
1 (Country)	US	1	
2 (State)	CA	11	1



Geography Level	Name	Source ID	Parent Source ID
3 (County)	Alameda	111	11
4 (City)	Pleasanton	1111	111
4 (City)	Dublin	1112	111

#### Creating and Scheduling the Import Activity

You can create an import activity, enter the import details, and schedule the import. An import activity includes selecting the source file or file location, mapping the source file to the database, and scheduling the import.

- 1. In the Setup and Maintenance work area, search for the **Manage File Import Activities** task. Click **Go to Task**.
- 2. In the Manage Import Activities page, click Create.
- 3. In the Create Import Activity: Map Fields page, map each field from your source file to the target object and attribute, as shown in the following 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. In the Create Import Activity: Map Fields page, map each field from your source file to the Oracle Sales Cloud database object and attribute, as shown in the following table.

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



Column Header	Example Value	Ignore	Object	Attribute
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 don't want to import a column in the text file, then you can select **Ignore**.

- Note: If you can't map the fields from your source file to the relevant target object, then see the import object spreadsheets.
- 6. Click Next.
- 7. In the Create Import Activity: Create Schedule page, select **Immediate** in the Schedule field so that the import will start as soon as you activate it.

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

8. Click Next.

#### Monitoring the Import Results

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

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

After the import activity has finished, the Status field value changes to Completed.

#### Related Topics

• File-Based Import Processing: How it Works

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

This topic explains how to prepare and import country structure data from an external data source 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 geography data compared to how the application represents the same data?
- Do you have to configure values in the application to map to your data values?
- Do you have to customize the application 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 the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a country structure.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country's structure, organized in a hierarchy using geography level numbers. For example, if you're 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 importing country structures, the application incorporates the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

#### Comparing Business Object Data

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

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you don't provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, 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
Country Structure	Country Structure Import Objects: How They Work Together

#### Extensible Attributes

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



#### Importing Country Structures Using File-Based Data Import

For the country structure business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data 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. When you're 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.

#### Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

## 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 objects for the Country Structure import object
- Target import object attributes
- Reference guide files for target import object attributes

## Country Structure Target Import Objects

The Country Structure import object contains one target import object. The target import object 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 Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs\_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A 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 prerequisite 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 Manage File Import Mappings task, or



you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

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

## Reference Files for Target Import Object Attributes

To access reference files for this object's target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs\_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

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 files that are available by target import object.

Target Import Object	Description	Reference Guide File Names
ImpGeoStructureLevel	Information that specifies a country's geography structure.	HZ_IMP_ GEO_STRUCTURE _ LEVELS_ Reference

#### Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Importing Country Structures Using File-Based Import: Quick Start
- Extending Oracle Sales Cloud: How It Works

## 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 application instance, and then after the territory geography zones are defined you can export the territory zones and import them into another application instance.

To define your territory geography zones and then import your territory zones into another application instance, you must complete the following steps:

- 1. Import the master reference geography data into the 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 application 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 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 application instance before you import the configuration package.

#### Related Topics

Managing Territory Geographies: Worked Example

## Defining Address Cleansing: Explained

Address cleansing validates, corrects, and standardizes address information that you enter in the application. Address cleansing, unlike geography validation, validates both the geography attributes and the address line attributes.

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

You can specify the real-time address cleansing level for each country by choosing either of these options:

- None: Specifies no real time address cleansing.
- Optional: Provides option to cleanse addresses.

Once you have enabled address cleansing for a country, a **Verify Address** icon appears at address entry points in the application. Click the icon to perform address cleansing and receive a corrected, standardized address. If the application does not find a matching address, then an alert message is displayed.

## FAQs for Define Geographies

## When do I define address cleansing?

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



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

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

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

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

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

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

#### Can I edit a specific geography in the geography hierarchy?

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

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

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

## Define Legal Jurisdictions and Authorities

## Jurisdictions and Legal Authorities: Explained

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

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

## Jurisdictions: Explained

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



#### Types of jurisdictions are:

- Identifying Jurisdiction
- Income Tax Jurisdiction
- Transaction Tax Jurisdiction

#### Identifying Jurisdiction

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

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

#### Income Tax Jurisdiction

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

#### Transaction Tax Jurisdiction

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

## Legal Authorities: Explained

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

For example, the Internal Revenue Service is the authority for enforcing income tax laws in United States. In some countries, such as India and Brazil, you are required to print legal authority information on your tax reports. Legal authorities are defined in the Oracle Fusion Legal Entity Configurator. Tax authorities are a subset of legal authorities and are defined using the same setup flow.

Legal authorities are not mandatory in Oracle Fusion Human Capital Management (HCM), but are recommended and are generally referenced on statutory reports.

## Creating Legal Jurisdictions, Addresses and Authorities: Examples

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

## Legal Jurisdictions

Create a legal jurisdiction by following these steps:

1. Navigator > Setup and Maintenance > Manage Legal Jurisdictions > Go to Task.



- 2. Select Create.
- 3. Enter a unique Name, United States Income Tax.
- 4. Select a **Territory**, United States.
- 5. Select a **Legislative Category**, Income tax.
- Select Identifying, Yes. Identifying indicates the first jurisdiction a legal entity must register with to do business in a country.
- Enter a Start Date if desired. You can also add an End Date to indicate a date that the jurisdiction may no longer be used.
- 8. Select a Legal Entity Registration Code, EIN or TIN.
- 9. Select a Legal Reporting Unit Registration Code, Legal Reporting Unit Registration Number.
- 10. Optionally enter one or more **Legal Functions**.
- 11. Save and Close.

#### Legal Addresses for Legal Entities and Reporting Units

Create a legal address for legal entities and reporting units by following these steps:

- Navigator > Setup and Maintenance > Manage Legal Address > Go to Task.
- 2. Select Create.
- 3. Select Country.
- 4. Enter Address Line 1, Oracle Parkway.
- 5. Optionally enter Address Line 2, and Address Line 3.
- **6.** Enter or Select **Zip Code**, 94065.
- 7. Select Geography 94065 and Parent Geography Redwood Shores, San Mateo, CA.
- 8. Optionally enter a Time Zone, US Pacific Time.
- 9. OK.
- 10. Save and Close.

#### Legal Authorities

Create a legal authority by following these steps:

- Navigator > Setup and Maintenance > Manage Legal Authorities > Go to Task.
- 2. Enter the **Name**, California Franchise Tax Board.
- 3. Enter the **Tax Authority Type**, Reporting.
  - Note: Create an address for the legal authority.
- 4. Select Create.
- **5.** The **Site Number** is automatically assigned.
- 6. Optionally enter a Mail Stop.
- 7. Select **Country**, United States
- **8.** Enter **Address Line 1**, 121 Spear Street, Suite 400.
- Optionally enter Address Line 2, and Address Line 3.
- 10. Enter or Select **Zip Code**, 94105.
- **11.** Select **Geography** 94105 and **Parent Geography** San Francisco, San Francisco, CA.
- 12. OK.
- **13.** Optionally enter a **Time Zone**, US Pacific Time.
- **14.** Optionally click the **One-Time Address** check box.
- 15. The From Date defaults to today's date. Update if necessary.
- **16.** Optionally enter a **To Date** to indicate the last day the address can be used.



- Note: You can optionally enter Address Purpose details.
- 17. Select Add Row.
- **18.** Select **Purpose**.
- **19.** The **Purpose from Date** will default to today's date.
- 20. Optionally enter a Purpose to Date.
- 21. OK.
- 22. Save and Close.

## Creating Legal Entities, Registrations, and Reporting Units: Examples

Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, and income, pay transaction taxes, or perform intercompany trading.

#### Legal Entity

Create a legal entity by following these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Entity > Go to Task.
- 2. Accept the default **Country**, United States.
- 3. Enter Name, InFusion USA West.
- 4. Enter Legal Entity Identifier, US0033.
- 5. Optionally enter **Start Date**. When the start date is blank the legal entity is effective from the creation date.
- 6. Optionally enter an End Date.
- Optionally, if your legal entity should be registered to report payroll tax and social insurance, select the Payroll statutory unit check box.
- 8. Optionally, if your legal entity has employees, select the Legal employer check box.
- **9.** Optionally, if this legal entity is not a payroll statutory unit, select an existing payroll statutory unit to report payroll tax and social instance on behalf of this legal entity.
- **10.** Enter the **Registration Information**
- **11.** Accept the default **Identifying Jurisdiction**, United States Income Tax.
- 12. Search for and select a Legal Address, 500 Oracle Parkway, Redwood Shores, CA 94065.

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

- 13. OK.
- **14.** Optionally enter a **Place of Registration**.
- **15.** Enter the **EIN** or **TIN**.
- 16. Enter the Legal Reporting Unit Registration Number.
- 17. Save and Close.
- 18. Navigator > Setup and Maintenance > Define Legal Entries > Manage Legal Entity > Select... to set scope.
- 19. Select the Manage Legal Entity.
- 20. In the \*Legal Entity list, select Select and Add.
- 21. Click Apply and Go to Task.
- 22. Select your legal entity.
- 23. Save and Close on the very bottom of the window.

This sets the scope for your task list to the selected legal entity.

24. Save and Close.



#### Legal Entity Registrations

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

- 1. Navigator > Setup and Maintenance > Manage Legal Entity Registrations: Verify that the Legal Entity scope value is set correctly.
- 2. Go to Task.
- 3. Select Create.
- 4. Enter Jurisdiction.
- 5. Enter Registered Address.
- 6. Enter Registered Name.
- Optionally enter Alternate Name, Registration Number, Place of Registration, Issuing Legal Authority, and Issuing Legal Authority Address, Start Date, and End Date.
- 8. Save and Close.

#### Legal Reporting Unit

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

- 1. Navigator > Setup and Maintenance > Define Legal Reporting Unit. > Manage Legal Reporting Unit. Verify that the Legal Entity scope value is set correctly.
- 2. Go to Task
- 3. Select Create.
- 4. Enter **Territory**, United States.
- 5. Enter Name.
- 6. Optionally enter a Start Date.
- 7. Enter Registration Information.
- 8. Search for and select Jurisdiction.
- 9. Enter Main Legal Reporting Unit information.
- 10. Select the value Yes or No for the Main Legal Reporting Unit. Set value to yes only if you are creating a new main (primary) legal reporting unit.
- **11.** Enter the **Main Effective Start Date**, 1/1/11.
- 12. Save and Close.

## Define Legal Entities: Manage Legal Entity

## Legal Entities: Explained

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

Legal entities have the following rights and responsibilities to:

- Own property
- Trade
- Repay debt



 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
- Minimizing the enterprise's tax liability
- 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, 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
- · 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), which 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. These registers are part of the Oracle Fusion Trading Community Architecture.



When your legal entities are trading with each other, represent them as legal entities and 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.

Several decisions you should consider when you create legal entities.

- The importance of using legal entity on 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 Using Legal Entities on 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, a sales order creates an obligation on the legal entity that books the order to deliver the goods on the acknowledged date. The creation also creates an obligation on 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.
- 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, Business Unit ACM UK has a default legal entity of InFusion UK Ltd. When a purchase order is placed in ACM UK, the legal entity InFusion UK Ltd is legally obligated to the supplier. Oracle Fusion Procurement, Oracle Fusion Project Portfolio Management, 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 wanted, 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 one segment 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 with 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. This solution is used 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 want to isolate the assets and liabilities for that entity.

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

If your enterprise regularly spins off businesses or holds managers accountable for utilization of assets, identify the business with a balancing segment value. If you account for each legal entity in a separate ledger, no requirement exists to identify the legal entity with a balancing segment value.

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 the portion is not a separate legal entity. If you do not map legal entities sharing the same ledger to balancing segments, you cannot distinguish them using intercompany functionality or track 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 feature to create intercompany entries automatically 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 then 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.

💡 Tip: In the Oracle Fusion Supply Chain applications, you can model intercompany relationships using business units, from which legal entities are derived.

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

## Define Legal Entities: Manage Legal Entity HCM Information

## **HCM Organization Models: Examples**

These examples illustrate different models for human capital management (HCM) organizations that include a legislative data group (LDG).

The example includes LDGs, which aren't organization classification, to show how to partition payroll data by associating them with a payroll statutory unit.

## Simple Configuration

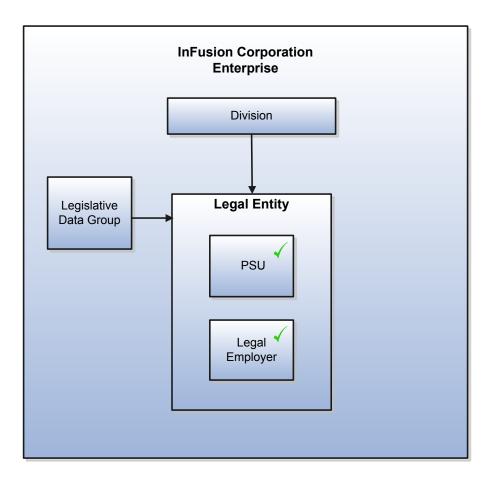
This example illustrates a simple configuration that does not include any tax reporting units.

Note the following:

- 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 these countries report at the legal entity level.



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



## Multiple Legal Employers and Tax Reporting Units

This example illustrates a more complex configuration. In this enterprise, you define one legal entity, InFusion US as a payroll statutory unit with two separate legal entities, which are also legal employers. This model shows multiple legal employers that are associated with a single payroll statutory unit. Tax reporting units are always associated with a specific legal employer (or employers) through the payroll statutory unit.

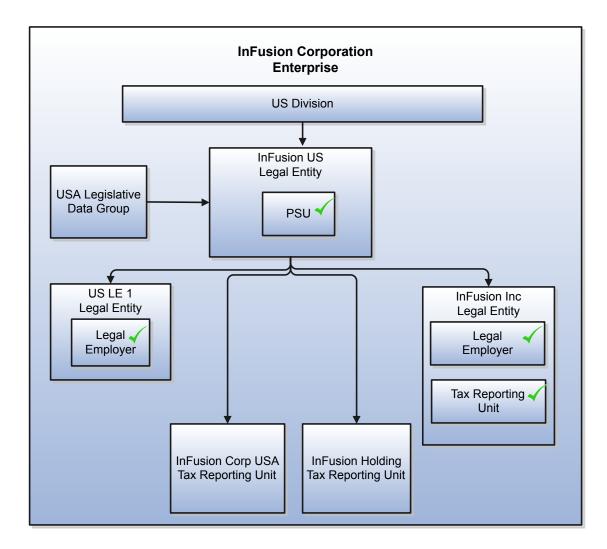
The implication is that payroll statutory reporting boundaries vary from human resources (HR) management, and you can categorize the balances separately by one of the following:

- Payroll statutory unit
- Legal employer
- 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.



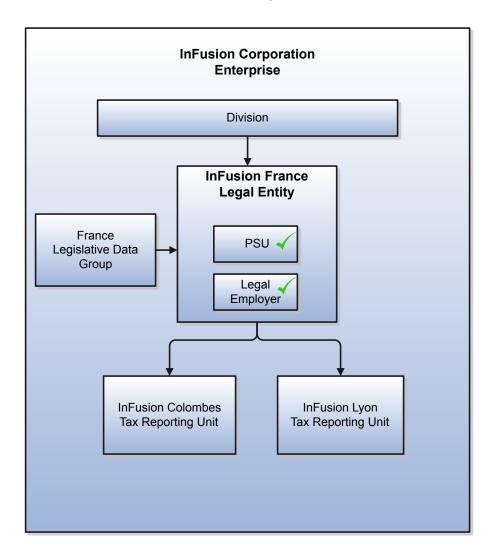
## One Payroll Statutory Unit and Two Tax Reporting Units

This model makes no distinction between a legal employer and a payroll statutory unit. You define tax reporting units as subsidiaries to the legal entity.

In this enterprise, legal entity is the highest level of aggregation for payroll calculations and reporting. Statutory reporting boundaries are 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. The legal entity 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

In this model, the enterprise has one legal entity. 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.

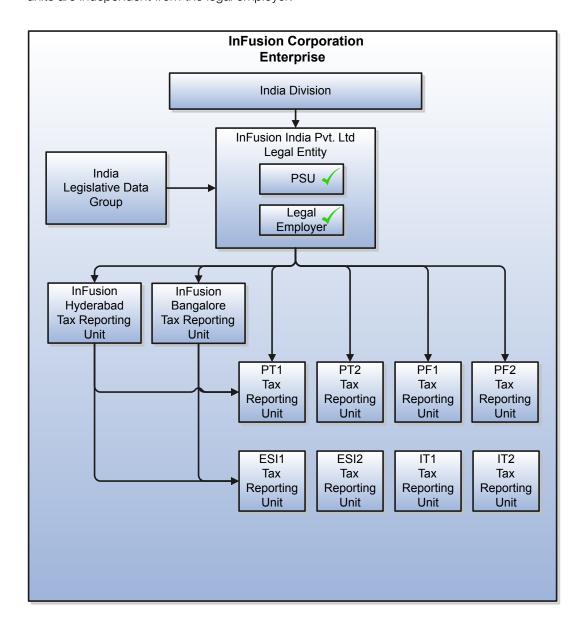
Using this model, you wouldn't typically:

- Report on tax reporting unit balances within a legal employer
- · Categorize balances 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. The tax reporting units are independent from the legal employer.



## Multiple Payroll Statutory Units with Several Tax Reporting Units

In this model, the enterprise has two legal entities. The 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.

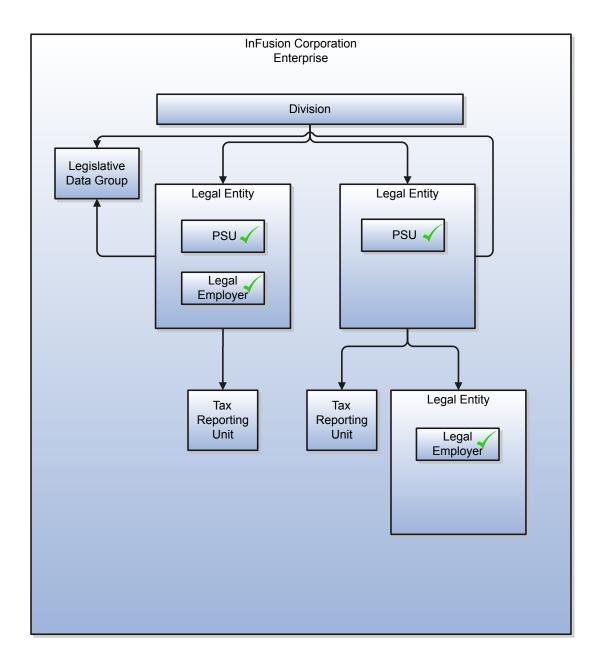
Using this model, you wouldn't typically:

- Report on tax reporting unit balances within a legal employer
- Categorize balances 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.



#### Related Topics

• Legislative Data Groups: Explained



# 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). 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.

Use the Manage Legal Reporting Unit HCM Information task to designate an existing legal reporting unit as a tax reporting unit. If you create a new legal reporting unit that belongs to a legal employer (that is not also a payroll statutory unit), you select a parent payroll statutory unit and then, when you run the Manage Legal Reporting Unit HCM Information task, you designate it as a tax reporting unit and select the legal employer.

#### Related Topics

• What's a 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 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, multiple company enterprise, then you register a payroll statutory unit in each country where you employ and pay people. You can optionally register a consolidated payroll statutory unit to pay and report on workers across multiple legal employers within the same country. You associate a legislative data group with a payroll statutory unit to provide the correct payroll information for workers.

## Define Legal Entities: Manage Legal Entity Tax Profile

## Party Tax Profiles: Explained

A tax profile is the body of information that relates to a party's transaction tax activities. A tax profile can include main and default information, tax registration, tax exemptions, party fiscal classifications, tax reporting codes, configuration options, and service subscriptions.

Set up tax profiles for the following parties involved in your transactions:

- First parties
- Third parties
- Tax authorities

#### First Parties

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

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

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

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

- Indicate that business unit tax setup is used and maintained based on the configuration of the associated legal entity at transaction time. The tax setup of the associated legal entity setup is either specific to the legal entity or shared across legal entities using the Global Configuration Owner setup.
- Indicate that tax setup is used and maintained by a specific business unit. Create configuration options for the
  business unit to indicate that the subscribed tax content is used for the transactions created for the business unit.

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



#### **Third Parties**

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

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

Note: You don't need to set up party tax profiles for third parties. Taxes are still calculated on transactions for third parties that don't have tax profiles.

#### Tax Authorities

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

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

#### Related Topics

- Specifying Third-Party Tax Profile Options: Points to Consider
- When does a party tax profile get created for a third party?

## Specifying First-Party Tax Profile Options: Points to Consider

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

The first-party tax profile consists of:

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



#### **Defaults and Controls**

The following table describes the defaults and controls available at the first-party tax profile level:

Option	Description
Set as self-assessment (reverse charge)	Automatically self-assess taxes on purchases.
Rounding Level	Perform rounding operations on the:
	<ul> <li>Header: Applies rounding to calculated tax amounts once for each tax rate per invoice.</li> <li>Line: Applies rounding to the calculated tax amount on each invoice line.</li> </ul>
Rounding Rule	The rule that defines how the rounding should be performed on a value involved in a taxable transaction. For example, up to the next highest value, down to the next lowest value, or nearest.
	Note: If you defined a rounding precedence hierarchy in the configuration owner tax option settings for the combination of configuration owner and event class, Oracle Fusion Tax considers the rounding details in the applicable tax profile.
Set Invoice Values as Tax Inclusive	This first party intends to send or receive invoices with invoice line amount inclusive of the tax amount.
	Note: This option overrides the tax inclusive handling setting at the tax level, but not at the tax rate level.

#### Tax Registrations

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

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

## Party Fiscal Classifications

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

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

## **Tax Reporting Codes**

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



#### **Configuration Options**

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

#### Service Subscriptions

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

#### Related Topics

Tax Registrations: Explained

Configuration Options: Explained

Setting Tax Reporting Configuration Controls for VAT: Critical Choices

Party Information: Explained

Rounding Precedence Hierarchy: How It's Determined

## FAQs for Manage Legal Entity Tax Profile

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

The legal entity party tax profile is automatically created when a legal entity record is created.

When a legal entity is created through a back-end process, a legal entity party tax profile is created, when you:

- Save a tax regime to which the legal tax entity subscribes.
- Save the configuration owner tax option that are defined for the legal entity.

You can also create a party tax profile for a legal entity manually. Use the Create Legal Entity Tax Profile page or edit the tax profile that was automatically generated with the relevant tax information.

## Define Legal Entities: Define Legal Reporting Units

## Planning Legal Reporting Units: Points to Consider

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



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

#### Planning Legal Reporting Units

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

## Define Business Units: Manage Business Units

## Business Units: Explained

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. A business unit can process transactions on behalf of many legal entities. Normally, it has 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 do the following:

- Assign your business units to one primary ledger. For example, if a business unit is processing payables invoices, then it must post to a particular ledger. This assignment is required for your business units with business functions that produce financial transactions.
- Use a business unit as a securing mechanism for transactions. For example, if you run your export business
  separately from your domestic sales business, then 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 provides the following advantages:

- Enables flexible implementation
- Provides consistent entity that controls and reports on transactions
- Shares 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 share reference data, such as payment terms and transaction types, across business units, or you can have each business unit manage its own set depending on the level at which you want 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 legal entities to ensure the uniqueness of sequencing.

In summary, use business units for:

Management reporting



- Transaction processing
- Transactional data security
- Reference data sharing and definition

#### Brief Overview of Business Unit Security

A number of Oracle Fusion Applications use business units to implement data security. You assign roles like Accounts Payable Manager to users to permit them to perform specific functions, and you assign business units for each role to users to give them access to data in those business units. For example, users which have been assigned a Payables role for a particular business unit, can perform the function of payables invoicing on the data in that business unit. Roles can be assigned to users manually using the Security Console, or automatically using provisioning rules. Business Units can be assigned to users using the Manage Data Access for Users task in Setup and Maintenance.

#### Related Topics

Reference Data Sets and Sharing Methods: Explained

# Define Business Units: Assign Business Unit Business Function

## Business Functions: Explained

A business unit can perform many business functions in Oracle Fusion Applications.

#### **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. In such cases, any segment that allows the identification of associated revenue and costs can be used as a profit center identification. The segment can be qualified as a cost center segment.

However, there are 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

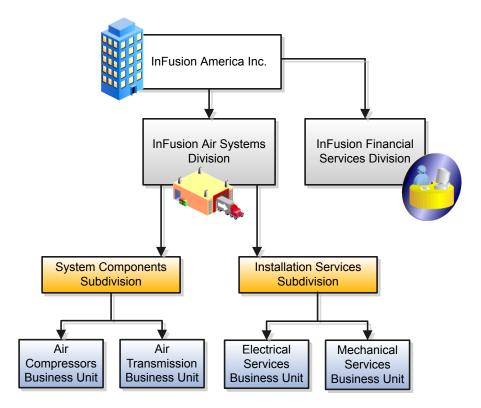
For example, your InFusion America Company provides:

- 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.

### FAQs for Assign Business Unit Business Function

Why did the newly added ledger and legal entity not show up in the list when assigning business functions?

When assigning the business functions, the list of values for the **Primary Ledger** and default **Legal Entity** are empty or do not show newly added values.

- Cause: The Review and Submit Accounting Configuration was not performed after ledger or legal entity were added.
- Solution: Submit the Review and Submit Accounting Configuration for the ledger from the Define Accounting Configuration task list in the Setup and Maintenance work area.



# Define Business Units: Manage Service Provider Relationships

## Shared Service Centers: Explained

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

#### Service Provider Model

The service provider model centralizes the following business functions:

- Procurement
  - Services business units that enable the Requisitioning business function.
  - Processes requisitions and negotiates supplier terms for client business units.
- Payables Payment
  - o Services business units that enable the Payables Invoicing business function.
  - Processes payments for client business units.
- Customer Payments
  - Services business units that enable the Billing and Revenue Management business function.
  - Processes payments for the transactions of client business units assigned the Billing and Revenue Management business function.

This functionality is used to frame service level agreements and drive security. The service provider relationships provides you with a clear record of how your business operations 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 for many business units.

#### Related Topics

Centralized Payment Processing Across Business Units: Explained

### 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 businesses. Your orders are segregated by business unit. However, all of these orders are managed from a shared service order desk in an outsourcing environment by your users who have access to multiple business units.

#### **Benefits**

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

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

## Service Provider Model: Explained

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

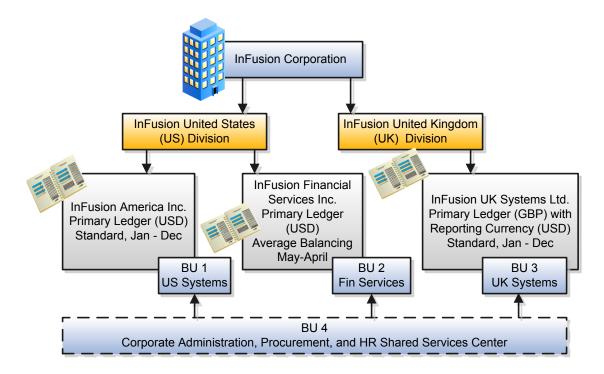
### Procurement Example

The Oracle Fusion Procurement product family has taken advantage of the service 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 takes care of supplier negotiations, supplier site maintenance, and purchase order processing on behalf of your client business units. Subscribe your client business units to the supplier sites maintained by the service providers, using a new procurement feature for supplier site assignment.

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



## Define Business Units: Specify Customer Contract Management Business Function Properties

### Customer Contracts Business Unit Setup: Explained

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



- setup task.
- Manage project billing options
- Set up clause numbering
- Set up the Contract Terms Library

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

**Note:** You must select a default currency in the customer or supplier business function properties page, if not populated automatically from the ledger assigned to the business unit in the assign business function

#### **Enabling Related Customer Accounts**

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

#### Managing Currency Conversion Options

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

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

In the Invoice Currency to Ledger Currency region:

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

### Managing Project Billing Options

The options available for selection in the Project Billing region control the behavior of project invoicing and revenue recognition for contracts with project-based work. Project billing can behave differently for external contracts (customer billing) or intercompany and interproject contracts (internal billing).

Set these options, which apply to all contracts:

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



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

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

Set this option only for customer billing:

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

#### Managing Clause Numbering

You can choose to number clauses manually or automatically.

If you choose the automatic numbering method, you must select a determinant level for the numbering. You must then select the appropriate clause sequence category from document sequences that you set up for this numbering level.

## Contract Terms Library Business Unit Setup: Explained

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

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

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

- 1. 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.
- 2. 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.

3. Specify the administrator who approves clauses available for adoption

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

#### Setting Clause Numbering Options

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

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

#### **Enabling Contract Expert**

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

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

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

The printed contract terms

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

The contract deviations report

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

#### Related Topics

- How the Selection of a Business Unit Affects Clauses and Other Objects in the Library
- Contract Expert: How It Works
- Contract Printing and Layout Templates: Explained

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



## Supplier Contracts Business Unit Setup: Explained

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

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

Note: The customer must select a default currency in the customer or supplier business function properties page, if not automatically populated from the ledger assigned to the business unit in the assign business function setup task.

#### Managing Contract Terms Library Setup Options

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

## Contract Terms Library Business Unit Setup: Explained

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

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

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

1. 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.

2. 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.

3. Specify the administrator who approves clauses available for adoption

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

#### Setting Clause Numbering Options

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

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

#### **Enabling Contract Expert**

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

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

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

• The printed contract terms

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

The contract deviations report

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

#### Related Topics

- How the Selection of a Business Unit Affects Clauses and Other Objects in the Library
- Contract Expert: How It Works
- Contract Printing and Layout Templates: Explained





## **5** Define Security

## Define Security: Overview

Oracle Enterprise Resource Planning Cloud (Oracle ERP Cloud) and Oracle Supply Chain Management Cloud (Oracle SCM Cloud) are secure as delivered; they limit access to one initial setup user. To enable application users to access application functions and data, you perform tasks in these task lists, as appropriate:

- Define Security for Financials
- Define Security for Procurement
- Define Users and Security for Product Management
- Define Security for Project Execution Management
- Define Security for Project Financial Management
- Define Security for Supply Chain Management

This topic introduces the tasks in these task lists. For more information on ERP and SCM security setup and task instructions, see these guides:

- Oracle Enterprise Resource Planning Cloud: Securing Oracle ERP Cloud.
- Oracle Supply Chain Management Cloud: Securing Oracle SCM Cloud.
- Note: You can perform most tasks in these task lists both during implementation, and later as requirements emerge.

## Manage Job Roles

The Oracle ERP Cloud and Oracle SCM Cloud security reference implementations provide many predefined job roles. You can perform the Manage Job Roles task to:

- Review the role hierarchy and other properties of a job or abstract role.
- Create custom job and abstract roles.
- View the roles assigned to a user.
- · View the users who have a specific role.

A user with the IT Security Manager or Application Implementation Consultant job role performs the Manage Job Roles task.

## Manage Duties

The Oracle ERP Cloud and Oracle SCM Cloud security reference implementations provide many predefined duty roles. You can perform the Manage Duties task to:

Review the duties of a job or abstract role.



- Manage the duties of a custom job or abstract role.
- Create custom duty roles.

A user with the IT Security Manager job role performs the Manage Duties task.

## Define Data Security

You can manage Oracle ERP Cloud and Oracle SCM Cloud application data by performing tasks in these task lists:

- Define Data Security for Financials
- Define Data Security for Procurement
- Define Data Security for Product Management
- Define Data Security for Project Financial Management
- Define Data Security for Supply Chain Management

You can perform the tasks in these task lists to:

- Manage data access sets that secure ledgers.
- Manage Human Capital Management (HCM) security profiles that facilitate data role assignment for application users.
- Manage data security policies that determine grants of entitlement to a user or role on an object or attribute group.

A user with the IT Security Manager job role performs the tasks in the Define Data Security task lists.

## Manage Role Provisioning Rules

You create role mappings to control the provisioning of all types of roles to application users by performing this task. For example, you can create a role mapping to provision the Accounts Payable Specialist role automatically to users that meet criteria specific to accounts payable users.

A user with the IT Security Manager job role performs the Manage Role Provisioning Rules task.

## Manage Oracle Social Network Objects

You can determine which business objects in Oracle ERP Cloud are available for social collaboration by performing this task. For example, use this task to enable discussion among Oracle Social Network users about requisitions, purchase orders, invoices, payments, receipts, and other transactions.

A user with the IT Security Manager or Application Implementation Consultant job role performs the Manage Oracle Social Network Objects task.

Note: You can perform this task after you set up and configure Oracle Social Network. If you do not use Oracle Social Network, you can skip this task.



## 6 Define Approval Management

## Approval Management: Highlights

Use approval management to define policies that apply to approval workflows. For example, to reflect your own corporate policies, you can specify levels of approval for expense reports over a particular amount and determine how the approvals are routed.

Approval management:

- Controls workflows for business objects such as expense reports.
- Enables you to define complex, multistage task routing rules.
- Integrates with the setup in Human Capital Management (HCM) to derive approvers based on the supervisory hierarchy.

To define approval management, use the Define Approval Management task list in the Setup and Maintenance work area. The task list includes setup tasks for managing workflow task configurations and approval groups.

### Task Configuration

Manage rule sets and rules that control approval flows.

- To configure a predefined approval policy, select the predefined rule set and click the Edit Task icon.
- To disable a predefined rule set, select the Ignore participant check box for that rule set.
- To edit the rules within a predefined rule set, you can insert, update, or delete while in edit mode.
- You can configure a specific rule to automatically approve a task without sending it 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.
  - Enter APPROVE in the Auto Action field.

#### Approval Groups

Each approval group includes a set of users that you configure to act on tasks in a certain pattern. Tasks can be defined to get routed to an approval group instead of an individual user.

- You can nest approval groups within approval groups.
- You have two options for defining the group:
  - Static: Select the specific users to include in the group.
  - o Dynamic: Provide the logic to use to determine the users in the group.



#### Customization

You can also customize predefined approval workflows, for example to add post-approval activities or additional stages (not available for Oracle Cloud implementations).

Refer to the Oracle Fusion Applications Extensibility Guide for Developers.
 See: Customizing and Extending SOA Components

## Other Workflow Setup

## Determining When Workflow Tasks Are Automatically Dismissed or Withdrawn: Points to Consider

Only workflow tasks with a final status, such as Completed or Withdrawn, can be purged and removed from users' worklists. Tasks go from the Assigned status to the Completed status when the final assignee approves or rejects the tasks, or, with **for your information** (FYI) tasks, when assignees explicitly dismiss the tasks. If assignees don't take actions that result in a final task status, within a certain period of time, then the tasks are automatically dismissed (FYI tasks) or withdrawn (all other tasks).

#### When Tasks are Eligible for Automatic Dismissal or Withdrawal

The FYI Notifications Expiration Period profile option determines when FYI tasks are eligible for automatic dismissal. In the Setup and Maintenance work area, use the Manage Applications Core Administrator Profile Values or Manage Administrator Profile Values task to set the profile option.

- Leave the profile option with the default value of 7, or replace it with a different number.
- The profile value represents the number of days after the FYI task is created.

When assignees don't read or dismiss an FYI task within the specified number of days after the task was created, the task is then eligible to be automatically dismissed.

All other tasks are eligible for automatic withdrawal when assignees don't take action to send the task to a final status within six months after the task was created.

#### When Eligible Tasks Are Automatically Dismissed or Withdrawn

Different processes run to automatically dismiss eligible FYI tasks or withdraw all other eligible tasks.

- FYI Tasks: The process runs every three days, starting the first day of each month. For example, it runs on May 1, 4, 7, and so on, and again on June 1 and every three days after. So, if you leave the FYI Notifications Expiration Period profile value at 7, then depending on when the process runs, an FYI task can be automatically dismissed within seven to ten days after it's created. The process changes the FYI task status from Assigned to Completed.
- **All Other Tasks:** The process runs every three days, starting the second day of each month. For example, it runs on May 2, 5, 8, and so on, and again on June 2 and every three days after. The process changes the status of eligible tasks to Withdrawn.

#### Related Topics

• Setting Profile Option Values: Procedure



## Setting Up the Worklist Region on My Dashboard: Points to Consider

Worklist: Notifications and Approvals is one of the predefined regions users can add to My Dashboard (**Navigator - My Dashboard**), which is blank by default. This region contains workflow tasks. To set up this Worklist region, select a value for the Welcome Dashboard Worklist Timeout Interval (ATK\_HOME\_PAGE\_WORKLIST\_TIMEOUT) profile option. In the Setup and Maintenance work area, use the Manage Application Toolkit Administrator Profile Values or Manage Administrator Profile Values task to set this profile option.

#### **Profile Value Considerations**

When users open My Dashboard and it contains the Worklist: Notifications and Approvals region, data for the region is retrieved. The profile option determines how long to continue retrieving before timing out and displaying no data.

- If you don't set a value for this profile option, which is blank by default, then the region doesn't time out.
- Retrieving data for the Worklist region affects the performance of My Dashboard as a whole. So, select a value for this profile option if your users have the Worklist region on My Dashboard and notice performance issues.

After the timeout, users can refresh the region to try retrieving the data again.

#### Related Topics

• Setting Profile Option Values: Procedure





## 7 Define Help Configuration

## Setting Up Help: Overview

Applications Help works without you having to set anything up. You can do the optional setup, mainly if you want to customize help. Select the help features you want, perform tasks in the Define Help Configuration task list, and customize help.

## Help Feature Choices

In the Setup and Maintenance work area, select help feature choices on the Features page when you configure your offerings. Feature choices determine:

- What's available in Applications Help
- What you can configure to set up help

The first feature choice for help is Local Installation of Help, and you must leave it selected. Other feature choices are:

- Access to Internet-Based Help Features
- Help Customization
- Custom Help Security

## Define Help Configuration Task List

In the Setup and Maintenance work area, use these tasks in the Define Help Configuration task list to configure Applications Help for all users:

- Set Help Options:
  - Determine if certain features of Applications Help are available to users.
  - Control how aspects of Applications Help work.
- Assign Help Text Administration Duty: Contact your security administrator to determine who can customize help.
- Manage Help Security Groups: Set up security to limit access to certain help files.

## Help Customization

After you configure help, you can review the predefined help and see if you want to add or customize any content. You can also customize help text that appears on the page, for example hints.

#### Related Topics

Features: Explained



- Help File Customization: Overview
- Customizing Help That Appears on the Page: Highlights

## Set Help Options

## Setting Up Access to Web Sites from Applications Help: Procedure

You can determine the Web sites that users can access from Applications Help.

#### Setting Up Access to External Web Sites

Follow these steps:

- 1. In the Setup and Maintenance work area, open the Features page for your offering.
- 2. Leave the Location Installation of Help feature choice selected.
- Select the Access to Internet-Based Help Features feature choice to allow access to Web sites from
  Applications Help. For example, some help files link to guides on the Oracle Help Center; this access is necessary for
  those links to work.
- 4. Select other feature choices as needed, and click **Done**.
- 5. Open the Set Help Options task.
- **6.** In the Web Sites Available from Help Site section, select the sites to link to from the Navigator menu in Applications Help.
- 7. Save your work.

### Setting Up Help Customization: Procedure

Users with the appropriate roles can customize predefined help or add their own files to help. To enable and set up help customization, do the following steps in the Setup and Maintenance work area, in the specified order.

#### Selecting Feature Choices

Perform these steps:

- 1. On the Features page for your offering, leave the Local Installation of Help feature choice selected.
- 2. Select the Help Customization feature choice.
- 3. Select the Custom Help Security feature choice if you want certain help files to be available only to a restricted set of users.
  - ▲ Caution: Don't select this feature choice if you don't have this requirement, because the feature can affect performance.
- 4. Save your work.

#### Setting Help Options

Perform these steps:

Open the Set Help Options task in the Setup and Maintenance work area.



#### 2. Optionally set options in these sections:

#### o Help Site Customization:

- Determine how users can identify custom files in Applications Help.
- Upload your own image to use as the background picture on the help home page. Select an image that's white along the entire left border, like you see in the default image.
- Oracle User Productivity Kit: Add a link in the Navigator in Applications Help to your custom User Productivity Kit library.
- **Privacy Statement:** Add a link to your own privacy statement. To see this link, users click their user name in the global area of Applications Help.
- 3. Save your work.

#### Providing Users Access to Help Customization

Only users with job roles containing the ATK\_CUSTOMIZE\_HELP\_TOPICS\_PRIV privilege can customize help. The Assign Help Text Administration Duty task is a reminder for you to follow up with your security administrator. Make sure that users who want to customize help have the access to do so.

#### Setting Up Help File Security

If you selected the Custom Help Security feature choice, then go to the Manage Help Security Groups task and select job roles to include in help security groups.

When you later customize a help file, you can select a group to determine which job roles have access to the file.

## FAQs for Set Help Options

#### Why can't I see certain sections on the Set Help Options page?

What's available on the page depends on the help feature choices that you select in the Setup and Maintenance work area. This table describes the correlation between feature choices and specific sections on the Set Help Options page.

Help Feature Choice	Section on Set Help Options Page		
Local Installation of Help	None, but without selecting this feature choice, you can't select the other help feature choices		
Access to Internet-Based Help Features	Web Sites Available from Help Site		
Help Customization	Help Site Customization		
	Oracle User Productivity Kit		
	Privacy Statement		
Custom Help Security	None		

#### When do I link to the Oracle User Productivity Kit library from Applications Help?

If you license Oracle User Productivity Kit and have custom User Productivity Kit content to share with your users. Topics that you add as custom help files in Applications Help are available only in the See It mode. However, in the library, users



can see the same topic in other modes. If you have User Productivity Kit versions earlier than 3.6.1, then you can't add User Productivity Kit topics as custom help. So the link to the library is the only way users can get custom User Productivity Kit content from Applications Help.

#### What's the URL for my Oracle User Productivity Kit library?

The full path from the Web server where you're hosting your Oracle User Productivity Kit content to the index.html file that
opens the table of contents for the library. For example, http://<your domain>.com/MyContent/PlayerPackage/index.html.

In this example, you or your administrator published one player package that contains all the content to be linked to from Applications Help, including the index.html file, and placed the PlayerPackage folder in a folder called MyContent on the Web server.

## FAQs for Assign Help Text Administration Duty

## Who can add and manage custom help?

Users with the Customize Help Topics (ATK\_CUSTOMIZE\_HELP\_TOPICS\_PRIV) privilege can customize:

- Help in Applications Help and help windows
- Pages in the Getting Started work area

This privilege is assigned by default to the administrators for product families. Your security administrator can define which users have job roles with this privilege.

## Manage Help Security Groups

## Creating Help Security Groups: Worked Example

This example shows how to create a help security group, which contains a set of job roles. You can later assign the help security group to particular help files so that only users with any of the included job 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 job role to the group.



#### **Prerequisites**

- 1. Open the Features page for your offerings in the Setup and Maintenance work area.
- 2. Make sure that the Location Installation of Help feature choice is selected.

#### Creating the Help Security Group

- 1. In the Setup and Maintenance work area, go to the Manage Help Security Groups 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.

To assign your new help security group to help files, you must create or edit help using the Manage Custom Help page, not help windows.

#### Related Topics

How can I restrict access to specific help files?





## 8 Define Application Toolkit Configuration

## Define Application Toolkit Configuration: Overview

Oracle Fusion Application Toolkit (ATK) provides many features that are available to users of all product families. These features include Applications Help, the Reports and Analytics pane, and the Watchlist. In the Setup and Maintenance work area, use the Define Application Toolkit Configuration task list to set up some of these components. Use the Define Help Configuration task list to set up 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.

#### **Tasks**

The Define Application Toolkit Configuration task list contains these tasks:

- Map Reports to Work Areas: Determine what's available in the Reports and Analytics pane for specific work areas.
- Set Watchlist Options: Define settings that affect what's displayed in the Watchlist and how often items are refreshed.
- Manage Application Toolkit Administrator Profile Values: Set profile options to affect how some Application Toolkit features work.

#### Related Topics

- Setting Up the Worklist Region on My Dashboard: Points to Consider
- Setting Up Help: Overview

## Map Reports to Work Areas

## Setting Up the Reports and Analytics Pane: Procedure

You can find the Reports and Analytics pane in many work areas, and the analytics and reports you see in the pane depends on the work area. You can define what's available for a specific work area, by mapping reports from the business intelligence (BI) catalog to that work area. In this mapping context, reports refer to both analytics and reports. Your changes apply to all users who have access to the work area you're mapping.

#### Mapping Reports from Your Work Area

To map reports to the work area that you're in:

1. Click the **Edit Settings** icon in the Reports and Analytics pane.



You see all the reports that are currently mapped to your work area.

- 2. Click Select and Add.
- 3. Find the report in the catalog and select it.
- 4. Click OK.
- 5. To remove any mapping, select the report and click **Remove**.
- 6. Save your work.

#### Mapping Reports to Any Work Area

To map reports to any work area that you have access to:

- 1. Go to the Setup and Maintenance work area and open the Map Reports to Work Areas task.
- 2. Select the application of the work area you want to map to.
- 3. Select the work area.
- 4. Click **Search** and see all the reports that are currently mapped to that work area.
- 5. Click Select and Add.
- 6. Find the report in the catalog and select it.
- 7. Click OK.
- 8. To remove any mapping, select the report and click **Remove**.
  - Tip: Click **Synchronize** to remove all mappings to any reports that are no longer in the catalog. You synchronize all work areas, not just the one you're mapping.
- 9. Save your work.

#### Related Topics

Reports and Analytics Pane: Explained

## Setting Reports Up for Scheduling in the Reports and Analytics Pane: Procedure

You can set up reports as scheduled processes, which means users can submit them from the Scheduled Processes and other work areas. If you want users to also submit these scheduled processes from the Reports and Analytics pane, then you must configure properties for the corresponding reports.

#### Enabling a Report for Scheduling

To enable scheduling in the Reports and Analytics pane:

- 1. Select the report in the business intelligence catalog and click **Edit**.
- 2. Click Properties.
- 3. On the General tab in the Properties dialog box, enter the following fields:

Field	Value
Enterprise Scheduler Job Package Name	The path for the job definition, for example: / oracle/ apps/ ess/ <pre>cproduct family&gt;/ <pre>cproduct&gt;/</pre></pre>



Field	Value
Enterprise Scheduler Job Definition Name	The job definition name (not display name), for example: ABCDEFG

#### Related Topics

- · Setting Reports Up to Run as Scheduled Processes: Points to Consider
- Accessing Report Components to Customize: Points to Consider

## FAQs for Map Reports to Work Areas

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

Either no reports are currently mapped to the work area you select on the Map Reports to Work Areas page, or you don't have access to the reports that are mapped.

Similarly, when you're selecting a report to map, you can see only the reports that you have access to. Ask your administrator to either:

- Assign you roles that have access to the reports you want to map to work areas.
- Grant the Reports and Analytics Region Administration Duty to someone who already has access to those reports.

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

In the Edit Settings window, you might not see a currently mapped report because you don't have access to it.

Similarly, when you're selecting a report to map, you can see only the reports that you have access to. Ask your administrator to either:

- Assign you roles that have access to the reports you want to map to work areas.
- Grant the Reports and Analytics Region Administration Duty to someone who already has access to those reports.

## Set Watchlist Options

## Disabling and Enabling Watchlist Categories and Items: Points to Consider

You can disable or enable predefined Watchlist categories and items for all users. Use the Set Watchlist Options task in the Setup and Maintenance work area.

Ultimately, what users see in their own Watchlist would be the categories and predefined items that you enable in the Set Watchlist Options page:

Plus any saved searches that the user is using as Watchlist items



- Minus any categories or items that the user decides to hide using Watchlist preferences
- · Minus any items with no results found, if the user decides to hide such items using Watchlist preferences

#### Any Category or Item

When you disable any category or item, you also disable the processes that calculate the Watchlist item counts. These processes include creating data caches, performing security checks, calling services across domains, running queries, and so on.

#### Predefined Watchlist Items

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 isn't available for users to display in their Watchlist.
- The item is removed from any Watchlist where it's currently displayed.

If you disable a Watchlist category, then the category is not available for users to include in their Watchlist. All Watchlist items within the category are also disabled.

#### User-Created Saved Search Watchlist Items

A Watchlist item with the User-Created Saved Search type doesn't appear in the Watchlist. It controls the display of the **Manage Watchlist** button on pages with saved searches. If you disable this type of Watchlist item, then:

- The **Manage Watchlist** option isn't available on the corresponding page, so users can't use their own saved searches as Watchlist items.
- Any user-defined saved searches (from that page) already used as Watchlist items are removed from the users' Watchlist. The saved searches are still available for searching, but not for the Watchlist.

#### Watchlist Category

If you disable a Watchlist category, then:

- The category isn't available for users to include in their Watchlist.
- All Watchlist items within the category are also disabled.

#### Related Topics

- Creating Watchlist Items: Procedure
- Displaying and Hiding Watchlist Items: Procedure

## Refresh Intervals for Watchlist Items: Explained

All Watchlist items have a predefined refresh interval, which controls how often the query that calculates the count for a Watchlist item can run. Use the Set Watchlist Options task in the Setup and Maintenance work area to edit the interval values.

#### How the Refresh Works

The count for any Watchlist item gets refreshed as follows.

• When users open the Watchlist in the global area for the first time after signing in, the item is refreshed if five minutes have passed since its last refresh in an earlier session.



· When users open the Watchlist again, the item is refreshed only if the time since its last refresh is equal to or greater than the refresh interval.

#### User-Created Saved Search Watchlist Items

What you enter as the refresh interval for a Watchlist item of type User-Created Saved Search applies to all Watchlist items based on saved searches that users create for that item. For example, you set the refresh interval for the Corporate Card Transactions item to five minutes. Multiple users create saved searches for corporate card transactions and use those saved searches as Watchlist items in their own Watchlist. All of these Watchlist items would have a refresh interval of five minutes.

#### Related Topics

• Creating Watchlist Items: Procedure

## FAQs for Set Watchlist Options

#### How can I change predefined Watchlist category and item names?

Edit the standard lookup type that stores the predefined Watchlist category and item names.

- 1. In the Setup and Maintenance work area, go to the Manage Standard Lookups task.
- 2. Find the lookup type for the Watchlist category you want to edit. Lookup types for predefined categories end with WATCHLIST, for example EXM EXPENSES WATCHLIST.
- 3. Edit the lookup type meaning to change the category name.
- 4. To change item names, edit lookup code meanings.
- 5. Save your work.

#### Related Topics

Lookups: Explained

## Manage Application Toolkit Administrator Profile Values

## Setting Up the Mapping Service for Contextual Addresses: Points to Consider

A contextual address is marked with an orange triangle that users can click to display the address on a map. The Mapping Service for Contextual Addresses profile option determines the mapping service to use to display the map. Use the Manage Administrator Profile Values or Manage Application Toolkit Administrator Profile Values task in the Setup and Maintenance work area to set the profile option value.

#### Profile Option Default

By default, the Mapping Service for Contextual Addresses profile option has no value.

Caution: Until you enter a valid value for this profile option, users get an error when they try to open a map for any contextual address.



#### Profile Option Value

After you find and select the Mapping Service for Contextual Addresses profile option, enter a mapping service URL in the Profile Value column, for example:

- http://maps.google.com/maps?output=embed&q=
- http://maps.yahoo.com/maps\_result.php?q1=
- http://maps.live.com/default.aspx?where1=
- http://bing.com/maps/?v=2&encType=1&where1=

You can include parameters in the URL. For example, to avoid a locator box in Google Maps, add &iwloc=& to the URL. So, you would enter http://maps.google.com/maps?iwloc=&&output=embed&q= as the profile value.

#### Related Topics

Setting Profile Option Values: Procedure

## Setting Up the Worklist Region on My Dashboard: Points to Consider

Worklist: Notifications and Approvals is one of the predefined regions users can add to My Dashboard (**Navigator - My Dashboard**), which is blank by default. This region contains workflow tasks. To set up this Worklist region, select a value for the Welcome Dashboard Worklist Timeout Interval (ATK\_HOME\_PAGE\_WORKLIST\_TIMEOUT) profile option. In the Setup and Maintenance work area, use the Manage Application Toolkit Administrator Profile Values or Manage Administrator Profile Values task to set this profile option.

#### Profile Value Considerations

When users open My Dashboard and it contains the Worklist: Notifications and Approvals region, data for the region is retrieved. The profile option determines how long to continue retrieving before timing out and displaying no data.

- If you don't set a value for this profile option, which is blank by default, then the region doesn't time out.
- Retrieving data for the Worklist region affects the performance of My Dashboard as a whole. So, select a value for this profile option if your users have the Worklist region on My Dashboard and notice performance issues.

After the timeout, users can refresh the region to try retrieving the data again.

#### Related Topics

• Setting Profile Option Values: Procedure



## 9 Define Applications Core Configuration

## Define Applications Core Configuration: Overview

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

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

#### Related Topics

Maintain Common Reference Objects: Overview

# Importing Flexfields, Lookups, or Profile Values Using Web Services: Example

You can import lookups, profile option values, and extensible flexfields into an application using the import function on the application page. An alternative is to use the web service <code>FndManageImportExportFilesService</code> for the import task. The web service uses the following methods:

- LOOKUDS processLookupsUploadFiles
- Profile Values processuserProfileValuesFile
- Extensible Flexfields -processFlexDataFiles

### Example

Given below is an example of importing the profile values using the web service. To import lookups or extensible flexfields, initiate the web service in the same way as shown here, after replacing the methods and other relevant values in the code.

To import profile values:

- Perform the following substeps to generate keystore for the security policy "oracle/ wss11\_saml\_or\_username\_token\_with\_message\_protection\_service\_policy":
  - a. Generate keystore using the command keytool -genkeypair -keyalg RSA -alias mycompkey -keypass welcomel -keystore mycompclient-keystore.jks -storepass welcomel -validity 3600.
  - b. Open the WSDL using the URL http://<host>:<port>/fndAppCoreServices/
    FndManageImportExportFilesService?wsdl through a web browser, and get the public key <wsdl:service>/
    <wsdl:port>/<wsa:EndpointReference>/<wsid:Identity>/<dsig:keyInfo>/<dsig:X509Data>/
    <dsig:X509Certificate>. Then, enclose it with ---- BEGIN CERTIFICATE ---- and ---- END CERTIFICATE ---- and save it to a file by name cdrmpk.cer.



- **c.** Store the key information in the truststore using the command keytool -importcert -alias cdrmkey -file cdrmpk.cer -keystore mycompclient-keystore.jks -storepass welcome1.
- 2. Open the WSDL page using the browser of your choice.
- 3. Export and save the associated security certificates Verisign Class 3 Public Primary Certification Authority G5 and Verisign Secure Server CA G3.
- 4. Use the following command to import the saved certificates into the trust store of the client computer.

keytool -importcert -keystore <truststore> -storepass <truststorepassword> -file <file location where
the above certificate is stored> -alias <alias for certificate>

5. Run the following command to generate the JAX-WS proxy for the FndManageImportExportFilesService Web service.

C:\Program Files\Java\jdk1.7.0\_04\bin>wsimport -s "d:\wsimport\FndManageImportExport" -d "d:\wsimport \FndManageImportExport" http://<host>:<port>/fndAppCoreServices/FndManageImportExportFilesService?wsdl

- 6. Save the generated code as a JAR file and name it FndManageImportExportProxy.jar.
- 7. Use the following code to create another JAR file to initiate the web service:

```
package com.oracle.xmlns.oracle.apps.fnd.applcore.webservices;
import com.sun.xml.ws.developer.WSBindingProvider;
import java.io.File;
import java.io.IOException;
import java.util.List;
import java.util.Map;
import javax.xml.ws.BindingProvider;
import javax.xml.ws.WebServiceRef;
import javax.xml.ws.handler.Handler;
import oracle.webservices.ClientConstants;
import weblogic.wsee.jws.jaxws.owsm.SecurityPoliciesFeature;
// !THE CHANGES MADE TO THIS FILE WILL BE DESTROYED IF REGENERATED!
// This source file is generated by Oracle tools
// Contents may be subject to change
// For reporting problems, use the following
// Version = Oracle WebServices (11.1.1.0.0, build 130224.1947.04102)
public class FndManageImportExportFilesServiceSoapHttpPortClient
  @WebServiceRef
 private static FndManageImportExportFilesService Service fndManageImportExportFilesService Service;
 public static void main(String [] args)
  System.setProperty("javax.net.ssl.trustStore", "<location of truststore which is used in II) to import
  the certificates>");
  System.setProperty("javax.net.ssl.trustStorePassword", "<truststore password>");
  fndManageImportExportFilesService Service = new FndManageImportExportFilesService Service();
  SecurityPoliciesFeature securityFeatures =
 new SecurityPoliciesFeature(new String[] { "oracle/
wss11 username token with message protection client policy",
 FndManageImportExportFilesService
  fndManageImportExportFilesService =
fnd Manage Import Export Files Service \ Service.get Fnd Manage Import Export Files Service Soap Http Port Files Fort Files Service Soap Http Port Files Fort Files Files Fort Files Files Fort Files Fort Files Fort Files Files Files Files Files Files Files Fort Files Files
```



```
(securityFeatures);
// Add your code to call the required methods.
WSBindingProvider wsbp = (WSBindingProvider) fndManageImportExportFilesService;
Map<String, Object> requestContext = wsbp.getRequestContext();
requestContext.put(ClientConstants.WSSEC KEYSTORE TYPE, "jks");
// Provide location of 'mycompclient-keystore.jks' which was created during Step I)
requestContext.put(ClientConstants.WSSEC KEYSTORE LOCATION, "/home/user1/mycompclient-keystore.jks");
requestContext.put(ClientConstants.WSSEC KEYSTORE PASSWORD, "welcome1");
requestContext.put(ClientConstants.WSSEC_RECIPIENT_KEY_ALIAS,"cdrmkey");
//Provide user who is having permission to initiate the service
requestContext.put(WSBindingProvider.USERNAME PROPERTY,"<username>");
requestContext.put(WSBindingProvider.PASSWORD_PROPERTY, "<password>");
String id = invokeUploadFiletoUCMMethod(fndManageImportExportFilesService);
if (id != null) {
invokeUserProfileValuesDataLoader(fndManageImportExportFilesService, new Long(id));
static String invokeUploadFiletoUCMMethod(FndManageImportExportFilesService
fndManageImportExportFilesService) {
String response = null;
DocumentDetails document = new DocumentDetails();
ObjectFactory objfactory = new ObjectFactory();
document.setFileName(objfactory.createDocumentDetailsFileName("import_data.txt"));
// Provide UCM repository - if repository is fin/tax/import then suffix each value with $ as mentioned
below
document.setDocumentAccount(objfactory.createDocumentDetailsDocumentAccount("fin$/tax$/import$"));
document.setDocumentTitle(objfactory.createDocumentDetailsDocumentTitle("VS"));
document.setContentType(objfactory.createDocumentDetailsContentType("plain/text"));
try {
// Provide location of bulk import data file in prescribed format
byte[] content =
org.apache.commons.io.FileUtils.readFileToByteArray(new File("/home/user1/import data.txt"));
//System.out.println("File content:" + new String(content, "UTF-8"));
document.setContent(objfactory.createDocumentDetailsContent(content));
} catch (IOException e) {
System.out.println(e.getMessage());
catch(Exception e) {
System.out.println("Exception: "+e.getMessage());
try {
response = fndManageImportExportFilesService.uploadFiletoUCM(document);
System.out.println("Response: " + response);
} catch (ServiceException e) {
System.out.println(e.getMessage());
return response;
```



```
static void invokeUserProfileValuesDataLoader(FndManageImportExportFilesService
fndManageImportExportFilesService,
Long id) {
String response;
try {
  response = fndManageImportExportFilesService.processUserProfileValuesFile(id);
  System.out.println("Response: " + response);
} catch (ServiceException e) {
  System.out.println(e.getMessage());
}
}
}
```

- Note: Wherever applicable, replace the values referring to the profile values with those of the lookups or flexfields value sets.
- 8. Save the generated output as a JAVA file and name it FndManageImportExportFilesServiceSoapHttpPortClient.java.
- 9. Use the JAVA file to build a JAR file, and name it FndManageImportExportClient.jar.
- 10. Use the following command to run the web service:

java -cp ./FndManageImportExportProxy.jar:./ws.api\_1.1.0.0.jar:./FndManageImportExportClient.jar FndManageImportExportFilesServiceSoapHttpPortClient

## **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	М	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 must be system or extensible to prevent deletion.

Once the customization level is set for a lookup type, it can't 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 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 setdeterminant

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, open the Setup and Maintenance work area, and use the tasks in the Define Lookups task list.



## Translating Lookups

You can translate the lookups that you defined to the preferred language(s) without changing the language session of the application. Use the translation option available on the lookup code table. By default, for each lookup, all the allowed language rows in the translator dialog box appear in the source language (the current session language). When you edit a particular language entry, you can modify the translated meaning and description to the language in which you want the lookup to appear. Once the updates are made, the end-users can view the lookup in the translated text.

Note: You can add the translation for only as many languages as are permitted by the administrator. The functionality to limit the number of languages displayed on the dialog box is controlled through the Translation Editor Languages profile option. It can be set at the SITE or USER level. If nothing is specified, all active languages are displayed.

#### Related Topics

Modules in Application Taxonomy: Explained

# Managing a Standard Lookup: Example

Creating a new standard lookup involves creating or selecting a lookup type containing the lookup code. The task also involves determining appropriate values for the lookup codes and their meanings. You can only create or edit 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 Oracle Application

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	Proceed	Yes	3



Lookup Code	Meaning	Enabled	Display Sequence
YELLOW	Caution	Yes	2

## The Resulting Data Entry List of Values

The enabled lookup codes appear in the list of values for the COLORS lookup type. You must select one of them to complete the activity.

The meanings and the codes are listed here. They appear in the order of the defined display sequence.

Displayed Value	Hidden ID
Stop	RED
Caution	YELLOW
Proceed	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 alphabetic, unless you enter a number manually to determine the order of appearance. Number 1 indicates the first value that appears in the list. Only lookups that are enabled and active between start and end dates, are visible.

### The Transaction Table

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

Transaction number	User name	Status
1	Jane	RED
2	Bob	YELLOW
2	Alica	DLUE
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 setdeterminant for the lookup codes. You can only create or edit 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. Use the Manage Set Assignments task to define and manage reference data set assignments.

## Selecting a Reference Group for a Set-Enabled Lookup Type

Specify a reference group for a set-enabled lookup type to 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 different meanings.

Reference Data Set	Lookup Code	Lookup Meaning
US	RED	Red
US	YELLOW	Yellow
US	GREEN	Green
EU	RED	Rouge
EU	ORANGE	Orange

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



Reference Data Set	Lookup Code	Lookup Meaning
CHINA	CSRC	Pending China Securities Regulatory Commission Review
CHINA	PR	Needs President's Approval
COMMON	REQUESTED	Customer Request

As per the above example, when end-users place a contract on hold in the US business unit, the three reason codes in the US set are available. When placing a contract on hold in the China business unit, the two codes in the China set are available.

# Importing Lookups: Procedure

On each page pertaining to the tasks of managing the Standard, Common, and Set Enabled lookups, use the Import option to import the lookup type and lookup code information.

### Prerequisite

The separate files containing the lookup types and lookup codes are already available in the document repository of Oracle WebCenter Content.

## Importing Lookups

To import lookups:

- 1. Sign in to the application as an implementation consultant or an administrator.
- In the Setup and Maintenance work area, search for and open a relevant lookups task list or task that you can use to import lookups.
- 3. Under Search Results, from the Actions menu, select **Import**.

The Import Lookups dialog box appears.

- 4. Select the WebCenter Content account to which the files were uploaded.
- **5.** Enter the names of the separate files containing the lookup type and lookup code information. The names here must match with the names of the files uploaded to the selected account.
- **6.** Click **Upload**. The lookup details are imported.

Note: If the import fails, click the link to the log file on the confirmation dialog box and examine the cause of failure.

# File Format for Importing Lookups: Explained

To import lookups into an application, you create separate text files containing the lookup types and lookup codes and upload them to the Oracle WebCenter Content document repository. The files must follow a specific format, as described here. After the files are in the document repository, you can then import the lookup types and lookup codes into the application following the instructions in the Importing Lookups: Procedure topic.



While creating the file, adhere to the following guidelines:

- Use a vertical bar or pipe (|) as a delimiter between fields for both header and value rows.
- Set the file encoding to UTF-8 without the Byte Order Mark (BOM), as per the Oracle WebCenter Content specification.

The following sections contain details about the specific lookup types and codes.

### Prerequisite

You must have worked with lookups in Oracle Cloud applications.

## Standard and Common Lookups

The lookup types and codes are similar for standard and common lookups. To create a file containing the lookup types, include the following headers:

- LookupType: The lookup type.
- Meaning: The display name of the lookup type.
- Description: The description of the lookup type. This header is optional.
- ModuleType: The module with which the lookup type is associated.
- ModuleKey: The module code.

Here's a sample file that contains the header values at the top and lists two lookup types to be imported. For importing several lookup types, add more entries in the same format.

```
LookupType|Meaning|Description|ModuleType|ModuleKey
AFLOG_22APR_1|Log1|AFLOG_desc_1|APPLICATION|FND
PROD_22APR_2|Product1|PROD_desc_2|APPLICATION|FND
```

To create a file containing the lookup codes, include the following headers.

- Required headers:
  - LookupType: The lookup type.
  - LookupCode: The lookup code associated with the lookup type.
  - o pisplaysequence: The sequence position at which the lookup code appears in the list of values.
  - EnabledFlag: Indicates the status of the lookup code, whether it's enabled for display or not.
  - Meaning: The display name of the lookup code.
- Optional headers:
  - o StartDateActive: Beginning of the date range during which the lookup code is active and visible on the page.
  - EndDateActive: End of the date range during which the lookup code is active and visible on the page.
  - Description: Description of the lookup code.
  - Tag: Any tag associated with the lookup code that may be used for a quick reference or retrieval of information.

Here's a sample file that contains two lookup codes:

```
 LookupType | LookupCode | DisplaySequence | EnabledFlag | StartDateActive | EndDateActive | Meaning | Description | Tag TasK_22APR_1 | Code1_1 | 1 | Y | 11/12/2014 | 1/5/2015 | TASK_22Apr_1 | Task_desc_1 | Tag1_1 TasK_22APR_1 | Code1_2 | 2 | N | 1/1/2014 | 1/11/2015 | TASK_22Apr_2 | Task_desc_2 | Tag1_2 TASK_22APR_2 | code2_1 | 3 | N | 11/12/2012 | 1/7/2015 | TASK_22APR_2 | Task_desc_2 | tag2_1 | Task_desc_2 | Tag1_2 | Task_desc_2 | Task_desc_2
```



TASK\_22APR\_2|code2\_2|3|Y|11/12/2012|1/7/2015|TASK\_22qpr\_2\_2|Task\_desc\_2\_2|tag2\_2

## Set Enabled Lookups

To create a file containing the set enabled lookup types, include the following headers:

- LookupType: The lookup type.
- Meaning: The display name of the lookup type.
- Description: The description of the lookup type. This header is optional.
- ModuleType: The module with which the lookup type is associated.
- ModuleKey: The module code.
- ReferenceGroupName: Name of the reference group that contains the reference data set.

Here's a sample that contains two set enabled lookup types:

```
 \label{lookupType} $$ LookupType | Meaning | Description | ModuleType | ModuleKey | ReferenceGroupName CODE_22APR_1 | CODE_22apr_1 | Code_desc_1 | APPLICATION | FND | BU_APAC CODE_22APR_2 | CODE_22qpr_2 | Code_desc_2 | APPLICATION | FND | BU_APAC | CODE_22APR_2 | CODE_22qpr_2 | Code_desc_2 | APPLICATION | FND | BU_APAC | CODE_22APR_2 | CODE_22qpr_2 | Code_desc_2 | APPLICATION | FND | BU_APAC | CODE_22APR_2 | CODE_22qpr_2 | CODE_2qpr_2 |
```

To create a file containing the set enabled lookup codes, include the following headers.

- Required headers:
  - LookupType: The lookup type.
  - o LookupCode: The lookup code associated with the lookup type.
  - o **DisplaySequence**: The sequence position at which the lookup code appears in the list of values.
  - EnabledFlag: Indicates the status of the lookup code, whether it's enabled for display or not.
  - Meaning: The display name of the lookup code.
  - o SetName: Name of the reference data set.
- Optional headers:
  - o startDateActive: Beginning of the date range during which the lookup code is active and visible on the page.
  - o EndDateActive: End of the date range during which the lookup code is active and visible on the page.
  - Description: Description of the lookup code.
  - Tag: Any tag associated with the lookup code that may be used for a quick reference or retrieval of information.

Here's a sample file that contains the header values at the top and lists four set enabled lookup codes to be imported. For importing several lookup codes, add more entries in the same format.

#### Related Topics

- Files for Import and Export: Explained
- · Files for Import and Export: Points to Consider



• Uploading Files to WebCenter Content Server: Procedure

# FAQs for Define Lookups

### How can I access predefined lookups?

Search for predefined lookups using the Define Lookups task list:

- 1. In the Setup and Maintenance work area, search for the **Define Lookups** task list and expand it to view the tasks.
- 2. Open the task that corresponds to the lookups you are searching for.
- Enter any of the search parameters and click Search. If you don't know the lookup type or the meaning, use the Module field to filter search results.
- 4. Click a lookup type to view its lookup codes.
  - Tip: Click the Query By Example icon to filter the lookup codes.

#### Related Topics

Using Query By Example: Procedure

## How can I edit lookups?

On the Define Lookups page, you can edit the existing lookup codes of a lookup type or add new lookup codes. To open the page, navigate to the Setup and Maintenance work area and search for the Define Lookup task list.

The task list contains three tasks:

- Standard Lookups
- Common Lookups
- Set-enabled Lookups

Each task contains a predefined set of lookup types classified and stored as per the functionality. Open a task to search and edit the required lookup. However, you may not be able to edit a lookup if its customization level doesn't support editing.

# Why can't I see my lookup types?

Lookup types are classified using tasks that involve a group of related lookups, such as Manage Geography Lookups. Each task gives you access only to certain lookup types. However, the generic tasks provide access to all lookups types of a kind, such as common lookups associated with the Manage Common Lookups task.

If the lookup types in an application are available in the standard, common, or set-enabled lookups view, they are central to an application. However, lookup types defined for a specific application are managed using the task or task list for that application.

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

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

A table-validated value set may consist of values that are validated through a SQL statement, which allows the list of values to be dynamic.

Tip: You can define a table-validated value set on any table, including the lookups table. Thus, you can change a lookup type 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 the list is table-validated
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	Validation 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	Both administrators and end-users manage these, except system lookups or predefined lookups at the system customization level, which can't be modified.	Usually administrators maintain these, except some product flexfield codes, such as GL for Oracle Fusion General Ledger that the endusers maintain.

Both lookup types and value sets are used to create lists of values from which users select values.

A lookup type cannot use a value from a value set. However, value sets can use standard, common, or set-enabled lookups.

## What's a lookup tag used for?

A tag is an additional label attached to the lookup. Tags are user defined and can be grouped depending on the user's requirement to make search convenient and effective.

The same tag may be used across lookup categories. In such cases, tags are used as a refined search criterion to filter information across several groups and get the custom search result.

## How can I search for a specific lookup code?

Use the Query By Example functionality to sort through hundreds of lookup codes for a lookup type, and display a specific lookup code. Enter the first few characters of the lookup code value in any of the relevant fields to filter the records.

Note: The search functionality is case sensitive.

# Manage Messages

# Messages: Explained

Messages provide users with information about business or application errors or warnings.

Typically, messages inform the users about the following:

Missing or incorrect data



- Status of an application, page, or a business object
- Status of an ongoing process
- Result of a user action

Besides notifying users about the problem, messages provide guidance to users on taking corrective action. Messages also warn users about the consequences of a certain action.

Oracle provides a set of predefined messages that are stored in a message dictionary. To create additional messages or modify the existing ones, use the Manage Messages task in the Setup and Maintenance work area.

Note: Don't delete predefined messages unless you are sure that they aren't used anywhere.

## Message Dictionary

The message dictionary stores messages that the application requires at run time. Messages are predefined for specific applications and modules, but a few are common messages that can be used in any application or module.

When you create messages, use the message text and the following components to cover additional details addressing users and help desk personnel:

- User Details: A detailed explanation of the message short text meant for users.
- Administrator Details: Details of the identified problem meant for the help desk personnel. The end users don't see
  this text.
- Cause: An end-user version of the cause of error.
- User Action: Instructions to users for addressing the identified problem. Where there is no guidance for end users, they must approach the help desk.
- Administrator Action: Corrective action that help desk personnel must take to correct the problem. This information is not available to the end users.

# Message Types: Explained

All messages must be associated with a message type. You can select the message type based on the message severity.

The available message types are:

- Error
- Warning
- Information
- UI String

## **Error Messages**

Use the Error message to inform users about, for example, entering incorrect data or performing actions that trigger validation. Error messages also inform users how to correct the situation so that they can continue with their task.

For example: You cannot specify a task without specifying the project.

Error messages also tell users about any serious problem with the application or process, and when they must seek assistance from the help desk. Some error messages trigger incidents or logs and have a mechanism to notify the help desk automatically.



## Warning Messages

Use the Warning message type to inform users about an application condition or a situation that might require their decision before they can continue.

Warning messages:

- Describe the reason for the warning and potential consequence of the selected or intended user action.
- Can be either a question or a statement.

For example: You delete the primary user. Do you want to continue?

The message is usually followed by Yes and No buttons.

## Information Messages

The Information message type tells users about changes in the application, a page, or a business object. These messages aren't triggered by users, and they don't have to take any immediate action in response.

For example: No events have been started or processed for this employee.

Use the Information message type to communicate information that is neither an error nor a warning.

## **UI String Messages**

Use the UI string message type to store shorter messages such as UI prompts, titles, or translated text, in the message dictionary.

# Grouping Messages by Category and Severity: Explained

You can group messages by severity to internally define logging and classifying incident policies. You can group by category based on the functionality or program.

Category and severity values do not appear in logging entries, incidents, or on the UI.

Note: The values in both options are predefined lookups but you can customize them. However, the maximum size of this field is 30 characters.

In the Setup and Maintenance work area, search for the Manage Messages task and use it to group the messages while creating or editing messages.

# Group by Category

Use it to group messages that relate to one functionality, such as a scheduled process, together into one category. Select one of the predefined categories to enable automatic incident creation when the error message activates. By default, the following categories are available:

- Product: Issues related to product functionality, setup, and maintenance. Such messages are typically intended for functional administrators or product super users.
- **System**: Issues concerning the application, database, technology stack, and so on. Such messages are typically intended for technical users such as application administrators or database administrators.
- Security: Issues concerning permissions, access, compliance, passwords, and so on. Such messages are typically
  intended for security administrators.



## Group by Severity

This grouping attribute is very specific and indicates the severity of the message. You must set the severity to High to enable automatic incident creation for the message. The following are predefined values, but you can add more if required.

- **High**: Used for serious messages that completely stop the progress of an important business process or affect a large user community, and require help desk's attention. Use this option to enable implicit incident creation for the message.
- Medium: Used for less severe and more isolated messages.
- Low: Used when you can't determine whether the message has a negative impact on end users or business processes.

## Logging and Incidents

Select the **Logging Enabled** check box to include the UI message in the stored log file. To enable automatic incident creation when the error message appears on the UI, set the severity to High.

Incidents collect information about the application errors for which users may require assistance from help desk. An incident contains information about the state of the application at the time the problem occurred. Help desk can use the information in the incidents to resolve the problems.

You may examine the stored logs and incidents (not available in Oracle Cloud implementations). To understand more about incidents and log files, refer to the Oracle Fusion Applications Administrator's Guide.

#### Related Topics

- What's an incident?
- Diagnostic Tests: Examples
- Diagnostic Tests: Highlights

# Creating and Editing a Message: Procedure

You may edit predefined messages stored in the message dictionary or create custom messages.

In the Setup and Maintenance work area, search for and open the Manage Messages task.

# Creating a Message

To create a message, perform the following steps:

- 1. On the Manage Messages page, click the **New** icon.
- 2. On the Create Message page, enter details under each section.
- **3.** In the Message Properties section:
  - **a.** Enter a unique message name that helps you find your custom messages and avoid name conflicts with predefined messages. Use underscore as a separator if the name contains multiple parts.
  - **b.** Select the application and module to associate the message with.
  - **c.** Enter a unique number that can be used as an identifier for the message. Users can quote this number when they contact the help desk for assistance.



- Note: You may use any number between 10,000,000 and 10,999,999. This number range is allocated for your custom messages. At run time, this number appears along with the application code after the message text, for example FND-2774.
- **d.** In the Translation Notes field, enter a description of the message indicating its use.
- e. Select the relevant message type, category, and severity.
- f. Select the Logging Enabled check box to create incidents or logs when messages appear on the UI.
- 4. In the Message Text section:
  - a. In the Short Text field, provide the actual message text that appears on the page at run time.
    - The short text can include tokens that are placeholders for displaying dynamic values at run time. However, to support easy translation, keep the message length (including values of tokens) under 160 characters in American English.
  - **b.** In the User Details field, enter information for the users to know why the message appeared. You can also include information for the users to resolve the issue themselves.
    - If your Short Text component has tokens that expand the text beyond the 160-character limit, move that portion of text here.
  - **c.** In the Administrator Details field, provide a detailed technical explanation of the message. This field is only visible to the help desk.
  - d. In the Cause field, provide a concise explanation of why the message appears. This text is visible to the users.
    - This information is optional and is only applicable to messages of type Error and Warning. However, if you mention the cause, you must mention in the User Action field the action that users must take.
  - e. In the User Action field, enter the user action to guide the users with steps to respond to the message and complete the task.
  - f. In the Administrator Action field, provide information that the help desk can use to resolve the problem.
- 5. In the Message Tokens section, define tokens that you want to use in this message.
- 6. Click Save and Close.

## Editing a Message

You may edit a predefined message or a custom message that you created.

To edit a message, search for a message on the Manage Messages page and perform the following steps:

- 1. Select the existing message and click the **Edit** icon.
- 2. On the Edit Message page, modify the existing details as per the instructions provided in the Creating a Message procedure.
  - Note: Don't edit the message number for predefined messages.
- 3. Click Save and Close.

#### Related Topics

How can I make message components visible only to specific users?



# Using Tokens in Messages: Points to Consider

Certain messages must display variable information at run time to help users clearly relate to the actual data and perform the required action. You can use tokens to contain variable values at run time, instead of writing a unique message for every possible situation.

Tokens are programmatic parts of message text that are placed within curly brackets when creating the message. Tokens serve as placeholders for the actual data. At run time, tokens dynamically display the actual text or value in the message, making a message specific to the situation. For example, the message "Enter an effective date that is the same as or later than {MATURITY\_DATE}" contains the token {MATURITY\_DATE}. At run time, instead of the token, the represented value (the actual date) appears. Thus, users see the message "Enter an effective date that is the same as or later than 25-APR-2015".

Use the Manage Messages task in the Setup and Maintenance work area to create and manage tokens. You must edit a message to define tokens for it. You can create tokens for a message and also delete them. However, you can't edit or delete the predefined tokens.

### **Token Definition**

To define a token, you must provide the following information:

- A unique name for the token.
- The type of data that the token replaces at run time. Available types are Date, Number, or Text.
- A description about what the token represents at run time.

#### Guidelines

Follow these general guidelines while defining tokens:

- Use curly brackets and all uppercase letters for the token names.
- Use underscore as a separator for a name containing two words or more.
- Don't use a space between words.

Follow these specific guidelines for each token data type.

Data Type	Guideline
Text	Use tokens for substituting any variable text element that qualifies as a noun.
Number	Plan carefully while using tokens for numbers especially, where a token could refer to either a singular or a plural number. You can use tokens for numbers representing an order, customer, or any other business object bearing a numeric value.
Date	Clearly define the context of the date, such as the start date, or end date, or a date range.

# Common Messages: Points to Consider

Message names that begin with FND\_CMN are common messages. Each common message can appear in multiple places in any product family across Oracle Fusion Applications. For example, the FND\_CMN\_NEW\_SRCH message can be used



for any search to indicate that no results were found. Common messages of type error or warning are part of the message dictionary.

## Creating and Editing Common Messages

You can create custom common messages for use in multiple places. However, ensure that you follow the predefined naming convention and numbering series associated with the application or module.

Note: Don't use FND\_CMN as the prefix for your custom messages because all the predefined common messages begin with it.

Common messages can be used in any application. Therefore, 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, ensure that the text is generic and applies to the entire site of Oracle Fusion Applications implementation.

# How can I make message components visible only to specific users?

Use the Manage Administrator Profile Values task to determine the visibility of the message components. For the **Message Mode** profile option, set the profile value to either User or Administrator. Based on the set value, the administrator or user actions and details appear for the intended audience.

However, the message components are visible to the audience based on their predefined access rights. Anyone having a user level access can't view the Administrator message components. If you set the profile value to the administrators of a specific product, the message components are visible only to that specific audience.

Note: If you don't set any value to the profile option, the visibility of the message component is determined by the default profile option settings.

# Define Document Sequences

# Document Sequences: Explained

You can assign a document sequence number to each business document or business event to uniquely identify it. For example, you can assign a document sequence number to an invoice that gets generated in response to a purchase order. However, you must enable the document sequencing option for that business document or event to start assigning the number. A document sequence number is useful in tracking completed or failed transactions.

You can set up document sequencing in three different modes:

- Automatic
- Manual
- Gapless



Note: Plan your document sequencing carefully before you start applying sequence numbers. Avoid switching to a different mode 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 automatically when the document is generated. That unique number is stored in the database. You can set an initial value for the numbering sequence. Thereafter. the numbering is sequential by date and time of creation. If you don't provide an initial value, the application sets the default initial value as 1.

## Manual Sequencing

Use the manual sequencing mode to assign a unique number to each document before the document is generated. In manual sequencing, the numeric ordering and completeness of a transaction is not automatically enforced. As a result, users can skip or omit numbers when entering the sequence value. However, each time a user assigns a number, 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. Sequence numbers are not assigned to incomplete or failed documents. As a result, the sequence is maintained for all the successfully generated documents.

Additionally, you can control the gapless document sequencing by enforcing the Transaction Date Validation option. When enabled, this option checks for the transaction date of a particular document and assigns the sequence number accordingly, to chronologically maintain the documents. The sequence numbers and the transaction dates are chronologically correlated to prevent any mismatch of a new document sequence being assigned to an older document or vice versa.

Note: Use this type of sequencing only if necessary because it may affect the performance of the application and slow down transaction processing.

#### Related Topics

Modules in Application Taxonomy: Explained

# 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.



Note: Once a document sequence category is created, you can't 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. Therefore, you must first decide the appropriate document sequence to use for a set of documents. Before you begin, here are a few prerequisites:

- Determine beforehand the mode of document sequencing, because you can't switch to other types once a sequence is in use.
- Note details such as the document sequence and document sequence category, for later reference.
- Identify if there are any restrictions or configuration prerequisites.
- 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. The sequence definition never expires if you don't provide an end date. Among the several options used in creating and editing document sequences, the following options are functionally more important and therefore must 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 must 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 can't 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. 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.
- Select only those tables that belong to the application associated with the category.
- Once a category is defined, you can't switch to another 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, 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 don't 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 determinant type.



At run time, when users create documents, the document sequence to be assigned is determined based on the following:

- An active assignment that matches the correct combination of category
- The numbering method
- 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 can't 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.

#### Related Topics

Managing Modules in Application Taxonomy: Points to Consider

# **Define Trees**

## Trees: Overview

Trees are hierarchical data models that you can use to organize data, apply business rules, control data access, and improve performance while querying. For example, an application maintains data of an organization called Vision Corporation that has two departments: Marketing and Finance. The Finance department has two functional divisions: Receivables and Payables. You can define a tree for Vision Corporation to establish a hierarchy across its departments, and their respective functional divisions. You can use the hierarchy to manage data at various levels of the organization.

In the Setup and Maintenance work area, search for and use the Manage Trees and Tree Versions task to organize data into hierarchies.

# Tree Structures

As the name suggests, tree structures provide you the framework to organize data such that you can establish a hierarchy for use by the tree. So, similar to a template, a tree structure guides the creation of a tree.

## Tree

A tree is an instance of the tree structure. The root node is the topmost nodal point of a tree. Child nodes branch off from the root node. Child nodes at the same level, branching off from a common parent node, are called siblings. Leaves are details branching off from a node but not extending further down the tree hierarchy. You can create trees for multiple data sources and share them across applications.

## Tree Versions

A tree by default has only one version. If required, you can create and maintain more than one editable tree version. At any point, only one tree version must be active. If you edit an existing version, it changes from active to draft. To use it again, you



must set it to active. Similar to any other version control system, versions of trees are maintained to track all the changes that a tree undergoes in its life cycle.

### Tree Labels

Tree labels are short names given to trees and tree structures. You can label the tree versions for better accessibility and information retrieval. When nodes are created in a tree, the existing tree labels are automatically assigned to the new tree nodes. You can use any table to store the labels and register the label data source with the tree structure.

In the Setup and Maintenance work area, search for and use the following tasks to work with trees:

- Manage Tree Structures: To create and update tree structures. You must first define a tree structure to create a tree.
- Manage Trees and Tree Versions: To create and update trees and their versions.
- Manage Tree Labels: To create and update tree labels.

# 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

While creating a tree structure, you must specify important details on the Create Tree Structure: Specify Definition page. As the source of the tree structure, you may either select the predefined tree structures and proceed with the definition or create custom tree structures.

#### Tree Node Selection

The data in **Tree Node** table maps to the data in nodes of the tree structure. You must select the correct and most appropriate tree node table to define the tree structure, based on which you establish the tree hierarchy. This selection also affects the level of security that is set on a tree node and its child entities.

#### Tree Sharing Mode

Use the following options 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 is associated with a specific reference data set.

#### Customization

You can customize the predefined tree structures as well as those you create. However, customizing a predefined tree structure is restricted and permitted through additional privileges. Customization is limited to specific tree nodes and below in the tree hierarchy.



### Multiple Tree Versions

Although multiple tree versions can exist together, Oracle recommends only one version be active at any given time. However, if required, you can have more tree versions to be in the active state for the same date range. You can use this flexibility to select the tree version you want to implement.

## Managing Tree Structures: Points to Consider

You can create, edit, and delete tree structures. You can also change the status of a tree structure and audit the changes.

#### Creating and Editing Tree Structures

When you edit an active tree structure, the status of the tree structure and all associated trees and their versions changes to draft. To reuse a tree structure, create a copy of the tree without copying the associated trees and tree versions. After making changes, set the status again to active. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.

For information about working with the offering-specific predefined tree structures, refer to the relevant product documentation.

#### Status

When 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	In a modified state, or not yet in use.
Active	In use, indicating that one or more trees or tree versions are created from the tree structure.
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. To manually trigger an audit, select Audit from the Actions menu on the Manage Tree Structures page. The Tree Structure Audit Result table shows a list of validations that ran against the selected tree structure.

#### **Audit Validators**

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	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
Restrict By Set ID	Manage Tree Structures: Specify Data Sources	If you select the Reference Data Set check box for the Restrict Tree Node List of Values Based on option, each of its data source view objects must have a reference data set attribute.	Even when the check box is selected, one or more data source view objects doesn't 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.
Available Label Data Sources	Manage Tree Structures: Specify Data Sources	If you select a list item from <b>Labeling Scheme</b> to specify a labeling scheme, the label data source view object specified for each data source must be accessible. Also, the primary keys must be valid. This restriction doesn't apply if you select <b>None</b> from the list.	<ul> <li>Any of the specified label data source view objects doesn't exist.</li> <li>Any of the specified label data source view objects doesn't have primary keys.</li> <li>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.</li> </ul>	Correct the specified label data source view object.  Correct the primary keys of the specified label data source view object.  Do one of the following:  Correct the primary keys in the label data source view object to match the primary keys that were earlier registered in FND_TS_DATA_SOUR  Correct the primary keys registered in that table to match the new view object definition.
Row Flattened Table Name	Manage Tree Structures: Specify Performance Options	You must specify a valid row flattened table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE_RF or a custom table.	<ul> <li>The specified table doesn't exist in the database.</li> <li>The specified table doesn't contain the same columns as the FND_TREE_NODE_R table.</li> </ul>	Correct the row flattened table definition.
Available Data Sources	Add Data Source	Each data source view object specified for the tree structure must be accessible, and all its primary key attributes must be valid.	Any of the specified data source view objects doesn't exist.	<ul> <li>Correct the specified data source view object.</li> <li>Correct the duplicate column</li> </ul>



Validator	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
			When you define a data source view object, keep the Use non-defined primary key columns check box deselected. The database automatically registers the primary keys for the view object. Select this check box if you want the database to register the primary keys you specify. However, 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 doesn't match the primary keys defined in the corresponding data source view object.      Any common attribute that exists in both the data source view object isn't of the same data type in both view objects.	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	Manage Tree Structures: Specify Performance Options	You must specify a valid column flattened table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE_CF or a custom table.	<ul> <li>The specified table doesn't exist in the database.</li> <li>The specified table doesn't contain the same columns as the FND_TREE_NODE_C table.</li> </ul>	Correct the column flattened table definition.
Restrict by Date	Manage Tree Structures: Specify Data Sources	If you select the <b>Date</b> Range check box for the <b>Restrict Tree Node</b> List of <b>Values Based</b>	Even when the check box is selected, one or more of its data source view objects doesn't contain effective start	If the date restriction is required for this tree structure, include the effective start date and effective end date



Validator	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
		on option for a tree structure, each of its data source view objects must have effective start date and end date attributes. This validation doesn't take place when the check box isn't selected.	date and end date attributes.	attributes on all data sources. Otherwise, deselect the check box.
Tree Node Table Name	Manage Tree Structures: Specify Definition	You must specify a valid tree node table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE or a custom table.	<ul> <li>No table is specified in the Tree Node Table field.</li> <li>The specified table doesn't exist in the database.</li> <li>The specified table doesn't contain the same columns as the FND_TREE_NODE table.</li> </ul>	Correct the tree node table definition.

## 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 ADF business components view objects, which are defined by application development.

### Labeling Schemes

Selecting a labeling scheme determines how the tree nodes are labeled. You may select a labeling scheme to assign at the data source level, at the parent node level, or keep it open for customers assignment. You may also choose not to have any labeling scheme. However, if you decide to use any of the labeling schemes, 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 Oracle ADF business components are used as data sources. To associate the view object with the tree structure, you can pick the code from Oracle 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
- Bl 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 <code>fnd\_tree\_node\_rf</code> 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 runtime 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 <code>fnd\_tree\_node\_cf</code> 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 Object

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.



### BI View Object

View objects from Business Intelligence can 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 BI view object generated by the tree management for business intelligence reporting and usage The BI 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.

### Search View Object

Specify the full name of the view object for the tree node to ensure that search operations performed on the tree node are efficient.

# 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.
	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. 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. You can also select and copy the associated tree 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. Each tree must contain a version.

Editing an existing tree provides you with the option of updating 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 canceled.

### 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.



Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
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 doesn't 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 <b>Allow Duplicates</b> check box isn't selected, the tree version must 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 must be valid and available in the underlying data source.	<ul> <li>A node in the tree version doesn't 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.</li> <li>The tree version contains a tree reference node, which references another tree version that does not exist.</li> </ul>	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 don't 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 must 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.



Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
SetID Restricted Node	On the Manage Tree Structures: Specify Data sources page, if the <b>Set ID</b> check box is selected to enable the <b>Restrict Tree Node List of Values Based on</b> 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 doesn't 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 <b>Labeling Scheme</b> list box, all nodes must have labels. This restriction doesn't apply when you select <b>None</b> from the <b>Labeling Scheme</b> list box.	The tree structure has a labeling scheme but the tree version has nodes without labels.	Assign a label to any node that doesn't have a label.
Date Restricted Node	On the Manage Tree Structures: Specify Data Sources page, if the <b>Date Range</b> check box is selected to enable the <b>Restrict Tree Node List of Values Based on</b> 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 doesn't apply if the check box isn't 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 isn't 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 isn't selected for the tree structure, there must not be more than one active tree version under a tree at any time. This restriction	Even when the check box isn't 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.



Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
	doesn't apply if the check box is selected.		
Range Based Node	On the Data Source dialog box, if the <b>Allow Range Children</b> check box isn't selected, range-based nodes are not permitted from that data source. This restriction doesn't apply if the check box is selected.	Even when the check box isn't 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 <b>Allow Use as Leaves</b> check box isn't selected, values from that data source can't be added as leaves (terminal nodes) to the tree version. This restriction doesn't apply if the check box is selected.	Even when the check box isn't 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 <b>Use All Values</b> option is selected to set the <b>Usage Limit</b> for the data source, every value in the data source must appear as a node in the tree. This restriction doesn't apply if <b>None</b> option is selected.	Even if the <b>Use All Values</b> option is selected, there are values in the data source that aren't in the tree version.	For each data source value that isn't yet available, add nodes to the tree version.

# Trees and Data Sources: How They Work Together

Data sources are the foundation of tree management. 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 and Data Storage

Tree structures contain the metadata of the actual data and the core business rules that manifest in trees and tree versions. You can 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, the 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 where a tree branches into levels. Nodes are the building blocks of a tree structure and are attached to tree versions. Whenever you create or edit a tree version, you need to specify its tree node.



In the Setup and Maintenance work area, search for the Define Trees task and access the Manage Trees and Tree Versions page.

## 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

Usually, the nodes at a particular level represent similar information. For example, in a tree that reflects the organizational hierarchy, all nodes representing divisions appear at one level and all the department nodes on another. Similarly, in a tree that organizes a user's product catalog, the nodes representing individual products might appear at one level and the nodes representing product lines on the immediate higher level.

The following node levels are in use:

- Root node: The topmost node in the tree structure
- Parent node: The node that branches off into other nodes
- Child node: The node that is connected to a node higher in hierarchy (parent node)
- Sibling node: Nodes that are at the same level and belong to the same parent node
- · Leaf node: Entities branching off from a node but not extending further down the tree hierarchy

## **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: Overview

Profile options are a set of preferences that you use to centrally manage the user interface settings and application behavior.

You can use the profile options to manage, for example:

- User preferences to specify language or currency.
- Configuration choices to change the user interface skin or appearance of fonts.
- Processing options to determine how much of an activity needs to be logged and at which level.

In the Setup and Maintenance work area, search for the **Define Profiles** task list. As an administrator or implementer, you can perform the following tasks:



Task Name	Function
Manage Profile Options	Create new profile options or modify existing profile options, except some which are predefined and restricted to prevent any modifications.
Manage Profile Categories	Group the profile options based on their functional similarities.
Manage Administrator Profile Values	Set the profile values for the enabled profile options to control application behavior.

For information on the predefined profile options, open the Setup and Maintenance work area, and use the Manage Profile Options task.

# Hierarchy in Profile Levels: Explained

The hierarchy in profile levels determines the context for making a profile option effective.

You can enable a profile option at the following levels:

- · Site level (lowest): The entire site of deployment
- Product level: A specific product component within the application
- User level (highest): A specific user

After you create or edit a profile option on the Manage Profile Options page, you must enable it. You can enable it at multiple levels. The setting at the highest enabled level takes precedence over the lower levels. User level is at the top of the hierarchy and always takes precedence over the settings at the product or site level.

On the Manage Administrative Profile Values page, set the profile value at any of the enabled levels of the profile option.

# Example of Profile Option Hierarchy

The following table shows an example of setting the currency profile option at different levels.

Profile Level	Hierarchy	Value Corresponding to the Selected Profile Level	Currency
Site	Lowest	NA	Euro
Product	Higher than Site	General Ledger	UK Pound Sterling
User	Highest	John Smith	US Dollar

For this example, both John Smith and Jane Lee are General Ledger users. However, based on the profile settings, John Smith would see US Dollar as the default currency, whereas Jane Lee would see UK Pound Sterling. Mary, who is neither a General Ledger user nor has any user-level preference defined, will see Euro as the default currency.



# Setting Profile Option Values: Procedure

Each profile option contains specific values that determine how it affects the application. You can add or modify the values for each profile option. Select or enter the value for one or more of the available levels (site, product, and user) so that each setting takes effect at the intended level.

## Setting the Profile Value

- 1. In the Setup and Maintenance work area, search for and open the Manage Administrator Profile Values task.
- 2. Search for and select the profile option.
- 3. In the Profile Values section, click Add. A new row is added for you to specify the following conditions:
  - Profile Level: Specify the level at which the profile value is to be set. If the profile value applies to the entire site, select Site.
  - Product Name: If you select **Product** as the profile level, select a product and specify the associated profile value.
  - User Name: If you select **User** as the profile level, select the user name and specify the associated profile value.
  - o Profile Value: Select or enter the value corresponding to the selected profile level.
  - Note: For an existing entry, you can modify only the profile value.
- 4. Repeat step 3 to add more rows and set the profile values.
- 5. Click Save and Close.
- Note: Changes in the profile values take effect for a user on the next sign in.

# Importing Profile Values: Procedure

Use the Import option on the Manage Administrator Profile Values page to import profile values in bulk and associate them with a profile option.

## Prerequisite

The file containing the profile values is available in the document repository of Oracle WebCenter Content.

# Importing Profile Values

To import profile values:

- 1. Sign in to the application as an implementation consultant or an administrator.
- 2. In the Setup and Maintenance work area, search for and open the **Manage Administrator Profile Values** task or a similar task used for importing profile values.
- 3. In the Profile Option: Profile Values section, from the Actions menu, select Import.
- 4. On the Import User Profile Values dialog box, select the WebCenter Content account to which the file was uploaded.
- **5.** Enter the name of the file containing the profile values. The name here must match with the name of the file uploaded to the selected account.
- **6.** Click **Upload**. The profile values are imported.



Note: If the import fails, click the link to the log file on the confirmation dialog box and examine the cause of failure.

# File Format for Importing Profile Values: Explained

To import profile option values into the application, you create a text file with the values and upload the file to the Oracle WebCenter Content document repository. The file must follow a specific format, as described here. After the file is in the document repository, you can then import the profile values into the application following the instructions in the Importing Profile Option Values: Procedure topic.

To create a file containing the profile values, include the following headers:

- ProfileOptionCode: The profile option code.
- LevelName: Must contain the value (Site, Product, or User).
- UserName: Must correspond to the registered user name in the application. Don't provide any other shortened or coded name of the user.
- ProfileOptionValue: The profile value to be imported.

While creating the file, adhere to the following guidelines:

- Use a vertical bar or pipe (|) as a delimiter between fields for both header and value rows.
- Set the file encoding to UTF-8 without the Byte Order Mark (BOM), as per the Oracle WebCenter Content specification.

Here's a sample file that contains the header values at the top and lists two profile values to be imported. For importing several profile values, add more entries in the same format.

ProfileOptionCode|LevelName|UserName|ProfileOptionValue AFLOG\_BUFFER\_MODE|USER|APP\_IMPL\_CONSULTANT|TEST AFLOG\_LEVEL|USER|APPLICATION\_DEVELOPER|FINEST

#### Related Topics

- Files for Import and Export: Explained
- Files for Import and Export: Points to Consider
- Uploading Files to WebCenter Content Server: Procedure

# Creating and Editing Profile Options: Procedure

Use profile options to centrally manage user preferences and control the general function of applications. For example, you can control user preferences involving language, date, time, currency, and other similar general settings.

You can create a profile option and also determine the level at which that profile option takes effect. You can also define the profile values for the profile option. The profile values appear on the Manage Administrator Profile Values page when you select the profile option.



## Creating a Profile Option

- 1. In the Setup and Maintenance work area, search for and open the Manage Profile Options task.
- 2. Click Actions New.
- 3. On the Create Profile Option page, fill all the fields with relevant details with specific attention to the following:
  - Use the SQL Validation field to provide an SQL statement that displays the permissible profile values to be used. Using an SQL statement, you can select the values from another table and display them as a list of values.

For example, to display the values **Yes** and **No** from a lookup table, you can use the following SQL statement:

select MEANING, LOOKUP CODE from FND LOOKUPS where LOOKUP TYPE='YES NO'

As a result, on the **Manage Administrator Profile Values** page, the profile values Yes and No are available for selection for that profile option.

- You can specify a date range to keep the profile option active during that period. Beyond the specified duration, the profile option automatically becomes inactive. If you no longer require the profile option, you must manually delete it from the **Manage Profile Options** page.
- 4. Click Save and Close.
- 5. On the Manage Profile Options page, search for the newly created profile option and from the results, select it.
- 6. In the Profile Option Levels section, do the following:
  - a. Under **Enabled**, select the levels at which you want to enable the profile option.
    - Note: You can enable a profile option at multiple levels, but a higher-level profile value overrides a lower-level value. Therefore, enable them only at the required levels.
  - b. Under **Updatable**, select the profile level at which you want implementors to have update privileges. Leave the check box deselected if you don't want the implementors to modify the profile values (they appear in readonly mode).
- 7. Click Save and Close.

To edit a profile option that you created, search for it and edit the necessary details.

# Managing Profile Categories: Points to Consider

You can create profile categories to group profile options based on their functional similarities and their use. In the Setup and Maintenance work area, search for the **Manage Profile Categories** task.

Profile categories help administrators or implementors in retrieving profile options using a search criterion on the **Manage Administrator Profile Values** page.

## Managing Profile Categories

Consider the following options while managing profile categories:

- Create profile categories and add existing profile options to them
- Add newly created profile options to existing custom profile categories



Note: While you can add a profile option to more than one category, some profile categories are predefined and restricted from any modifications. So, you can't edit them or add profile options to them.

## Setting Display Sequence for the Profile Options

You must set the display sequence for each profile option that you add to a profile category. Display sequence determines the order in which the profile options appear in a search result, based on the profile category. You can set the sequence beginning with zero or one for the first profile option to display, and proceed sequentially to assign the values to the remaining profile options.

The following table demonstrates the effect of the display sequence on the profile options when they are retrieved as search results.

Profile Category	Included Profile Option - Assigned Display Sequence	Display Sequence of Profile Options in the Search Results
Attachments	<ul><li>Attachment File Directory - 2</li><li>Indicate Attachments - 1</li></ul>	<ol> <li>Indicate Attachments</li> <li>Attachment File Directory</li> </ol>

# How can I access predefined profile options?

Search for predefined profile options using the Define Profiles task list:

- 1. In the Setup and Maintenance work area, search for the Manage Profile Options task and open it.
- 2. Enter any of the search parameters and click **Search**.
  - ▼ Tip: If you don't know the profile option code or the display name, use the Application or Module fields to filter search results.
- 3. Click a profile option to view its details.

# Define Flexfields

# Flexfields: Overview

A flexfield is an extensible set of placeholder fields associated with business objects and placed on the application pages. You can use flexfields to extend the business objects and meet enterprise data management requirements without changing the data model or performing any database programming. Flexfields help you to capture different data on the same database table.

For example, an airline manufacturer may require specific attributes for its orders that aren't predefined. Using a flexfield for the order business object, you can create and configure the required attribute.

Flexfields that you see on the application pages are predefined. However, you can configure or extend the flexfields, or modify their properties. Users see these flexfields as field or information attributes on the UI pages. To use flexfields, search for and



open the **Define Flexfields** task list in the Setup and Maintenance work area. You can use the following tasks contained within it:

- Manage Descriptive Flexfields: Expand the forms on the application page to accommodate additional information
  that is important and unique to your business. You can use a descriptive flexfield to collect custom invoice details on
  a page displaying invoices.
- Manage Extensible Flexfields: Establish one-to-many data relationships and make application data contextsensitive. The flexfields appear only when the contextual data conditions are fulfilled. Thus, extensible flexfields provide more flexibility than the descriptive flexfields.
- Manage Key Flexfields: Store information combining several values, such as a number combination. The key
  flexfields represent objects such as accounting codes and asset categories.
- Manage Value Sets: Use a group of values to validate the data entered in the flexfields.
  - Note: You can manage value sets within the Manage Descriptive Flexfields or Manage Extensible Flexfields tasks.

For more information about specific predefined flexfields, open the Setup and Maintenance work area, and use the tasks in the Define Flexfields task list.

# Types of Flexfields

The following three types of flexfields provide a means to customize the applications features without programming:

- Descriptive
- Extensible
- Key

### Related Topics

Modules in Application Taxonomy: Explained

# Flexfield Components: Explained

A flexfield is made up of several data entities that store and render information pertaining to flexfield configuration.

Flexfields are made up of the following components:

- Segments
- Value Sets
- Contexts
- Structures

# Segments

A segment is a field within a flexfield and represents a single table column of your database. When configuring a flexfield, define the appearance and meaning of individual segments. Segments represent attributes of information. Segments can appear globally wherever the flexfield is implemented, or based on a structure or context. Each segment captures a single atomic value and represents an attribute of information.



The characteristics of a segment vary based on the type of flexfield in which it's used.

- In key flexfields, a segment describes a characteristic of the entity. For example, a part number that contains details about the type, color, and size of an item.
- In a descriptive or extensible flexfield, a segment represents an information attribute on the application page.
   For example, details about a device containing components, some of which are global while the remaining are contextually dependent on the category of the device.

### Value Sets

Users enter values into segments while using an application. A value set is a named group of values that validate the content of a flexfield segment. You configure a flexfield segment with a value set to enforce entries of only valid values for that segment.

The configuration involves the following tasks:

- Defining the values in a value set, including characteristics such as the length and format of the values.
- Specifying formatting rules or values from an application table or predefined list.

Multiple segments within a flexfield, or multiple flexfields, can share a single value set.

### Contexts

Context-sensitive flexfield segments are available to an application based on a context value. You define contexts as part of configuring a flexfield. Users see global segments as well as any context-sensitive segments that apply to the selected context value.

In descriptive flexfields and extensible flexfields, you can reuse the context-sensitive segments that are based on the database columns, in multiple contexts.

# Structures

Key flexfields have structures. Each key flexfield structure is a specific configuration of segments. Adding or removing segments, or rearranging their order, produces a different structure. You can reuse the segments that are based on the database columns, in multiple structures.

# Configuring Flexfields: Overview

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

# Overall Process for Configuring Custom Attributes

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

- 1. Use the Highlight Flexfields feature from the Administration menu to find flexfields on pages associated with business objects.
- 2. Plan the flexfield configuration.
- 3. Plan flexfield validation.



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

A simple configuration is limited to such actions as adding a format-only field or adding a field with a basic list of values.

# Overall Process for Configuring Custom Keys

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

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

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

# Flexfields at Run Time: Explained

Business objects have an associated descriptive or extensible flexfield. Using these, you can create custom attributes for the business object at run time. Some business objects have an associated key flexfield for configuring flexible multiple part keys.

# Finding Flexfields on a Page

At run time, the custom attributes you define as flexfield segments appear in the application page just like any other attribute. However, each type of flexfield appears in a different way.

The following characteristics help you determine the type of flexfield on the application page:

 Descriptive flexfield segments appear as label and field pairs or as a table of fields that correspond to the column headers. The fields represent the flexfield segments and accept values that derive from 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.
- Key flexfields appear in the application page as a field with a key flexfield icon, where the field's value is a collection of segments.

To locate flexfields on a page, in the global area, select your user name and under the Settings and Actions menu, select **Highlight Flexfields**. The page renders in a special mode, displaying the location of flexfields, if any, on the page. Do the following:

- Hover over the **Information** icon to view flexfield details.
- Click the Configure Flexfield icon to manage the flexfield using the Manage Flexfields task.
- Click the Add Context Value, Add Segment, or Edit Segment icons to add a context value or edit a global or context-sensitive flexfield segment. This applies to both descriptive and extensible flexfields.
- Note: Not all flexfields are available for creating custom attributes. For example, some flexfields are protected, and you either can't edit their configurations at all, or can do only limited changes to them. Consult the product-specific documentation in Oracle Fusion Applications Help to verify whether there are any restrictions on using the flexfield.

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

When you no longer want to view the flexfields on a page, select **Unhighlight Flexfields** from the Administration menu.

# Customizing Flexfields Using Page Composer: Explained

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

In Page Composer, to customize:

- Extensible flexfields, open the page in Source view, and look for a region that is bound to an
  EffContextsPageContainer task flow. This is the container for the extensible flexfield attributes and contexts. To view
  the flexfield code and identifying information, open the properties panel for the region. To customize any component
  within the region, select the desired tag and click Edit.
- Descriptive flexfields, open the page in Source view, and look for <descriptiveFlexfield> elements. Open the properties panel for the element to view the flexfield code and identifying information. Within the properties panel, you may customize properties for the global and context-sensitive segments or re-order the segments on the page.

# Importing Extensible Flexfields: Procedure

Use the Import option on the Manage Extensible Flexfield page to bulk import the flexfield data that includes value set, context, and context segment details.

# Prerequisite

The files containing the flexfield details are available in the document repository of Oracle WebCenter Content.



# Importing Flexfields

To import flexfields:

- 1. Sign in to the application as an implementation consultant or an administrator.
- In the Setup and Maintenance work area, search for and open the Manage Extensible Flexfield task or a similar task for importing flexfields.
- 3. Under Search Results, from the Actions menu, select Import.
- 4. On the Upload Flex Data dialog box, select the WebCenter Content account to which the files were uploaded.
- **5.** Enter the names of the separate files containing the value set, context, and context segment information. The names here must match with the names of the files uploaded to the selected account.
- 6. Click **Upload**. The flexfield details are imported.
  - Note: If the import fails, click the link to the log file on the confirmation dialog box and examine the cause of failure.

#### Related Topics

• Importing Value Set Values Using Web Services: Procedure

# File Format for Importing Extensible Flexfields: Explained

To import extensible flexfields into an application, you create separate text files for the value sets, contexts, and context segments. Then, you upload them to the Oracle WebCenter Content document repository. Each file must follow a specific format, as described here. After the files are placed in the document repository, you can import the values sets, contexts, and context segments into the application following the instructions in the Importing Flexfields: Procedure topic.

While creating the file, adhere to the following guidelines:

- Use a vertical bar or pipe (|) as a delimiter between fields for both header and value rows.
- Set the file encoding to UTF-8 without the Byte Order Mark (BOM) as per the Oracle WebCenter Content specification.

The following sections contain specific details about each file format.

# Prerequisite

You must have worked with flexfields in Oracle Cloud applications.

### Value Sets

To create a file containing value sets, include the following headers:

- ValueSetCode
- ModuleType
- ModuleKey
- ValidationType
- ValueDataType
- ValueSubType
- MaximumLength



· Description - This is optional.

Here's a sample file that contains the header values at the top and lists three value sets to be imported. For importing several value sets, add more entries in the same format.

```
ValueSetCode|ModuleType|ModuleKey|ValidationType|ValueDataType|ValueSubtype|MaximumLength|Description
VS_TEST_91|APPLICATION|FND|INDEP|VARCHAR2|TEXT|2|desc1
VS_TEST_92|APPLICATION|FND|INDEP|VARCHAR2|TEXT|3|
VS_TEST_93|APPLICATION|FND|INDEP|VARCHAR2|TEXT|3|desc3
```

### Context

To create a file containing the contexts, include the following headers.

- · Required headers:
  - ApplicationId
  - EFFCode
  - ContextCode
  - Name
  - MultirowFlag
  - EnabledFlag
  - FlexfieldUsageCode
- Optional headers:
  - Description
  - ViewPrivilegeName
  - EditPrivilegeName
  - CExtAttribute1

Here's a sample file that contains the header values at the top and lists three contexts to be imported. For importing several contexts, add more entries in the same format.

```
ApplicationId|EFFCode|ContextCode|Name|EnabledFlag|MultirowFlag|Description|FlexfieldUsageCode|ViewPrivilegeName|CExtAttribute1|CExtAttribute2|CExtAttribute3|CExtAttribute4|CExtAttribute5
0|FLEX_SN_EFF1|OBJ_TEST_4|Object test 44|Y|N|desc 44 3363|FLEX_SN_EFF1_USAGE2|flex_sn_sitems_view|||||
0|FLEX_SN_EFF1|OBJ_TEST_3|Object test 33|Y|N||FLEX_SN_EFF1_USAGE1|flex_sn_view|||||NEW_TEST11
0|FLEX_SN_EFF1|OBJ_TEST_3|Object test 33|Y|N|new desc aug 14|FLEX_SN_EFF1_USAGE2|flex_sn_sitems_edit|
CE1 TESTupd aug15||||
```

# Context Segment

To create a file containing context segments, include the following headers.

- Required headers:
  - ApplicationId
  - EFFCode
  - ContextCode
  - SegmentCode
  - Name



- ColumnName
- ValueSetCode
- DisplayType
- Prompt
- ShortPrompt
- EnabledFlag
- RequiredFlag
- ReadOnlyFlag
- Optional headers:
  - Description
  - UOMclass

Here's a sample file that contains the header values at the top and lists a context segment to be imported. For importing several context segments, add more entries in the same format.

ApplicationId|EFFCode|ContextCode|SegmentCode|Name|ColumnName|ValueSetCode|DisplayType|Prompt|ShortPrompt|EnabledFlag|RequiredFlag|ReadOnlyFlag|Description|UomClass
0|FLEX\_SN\_EFF1|ColdSourceTargetContext|Cold\_TEST\_SEG1|cold\_seg\_1|ATTRIBUTE\_CHAR3|EFF\_BASIC\_FMT\_CHR|TEXT\_BOX|
Prompt|Short\_Prompt|Y|Y|N|TEST\_desc\_aug\_14\_3PM|

#### Related Topics

- Files for Import and Export: Explained
- Files for Import and Export: Points to Consider
- Uploading Files to WebCenter Content Server: Procedure

# Flexfields and Oracle Applications Cloud Architecture: How They Work Together

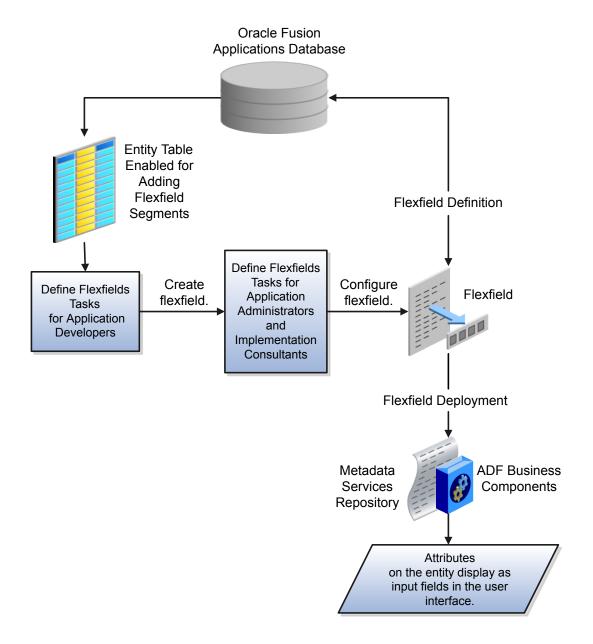
To capture additional data, administrators or implementors configure flexfield segments that represent attributes of business objects. Business objects are enabled for both descriptive flexfields and extensible flexfields.

The following figure shows the layers involved in configuring a flexfield:

- The business entity table and metadata in the database.
- The ADF business component objects. These are derived from the metadata and stored in Oracle Metadata Services (MDS) repository.
- The user interface where 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's available for configuration. Administrators and implementation consultants configure segments and other properties of the available flexfields. This information is stored as additional flexfield metadata in the database. Deploying the flexfield generates ADF business components based on the flexfield metadata in the database.

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

- Integration
- Deployment



- Import and export
- Run time
- Patching

# Integration

The attributes that you add by configuring flexfields are available throughout the Oracle Fusion Middleware technology stack. You can use the flexfield segment's Application Programming Interface (API) to identify segments and integrate the flexfields in the following:

- User interface pages
- Service-oriented Architecture (SOA) infrastructure
- Oracle Business Intelligence
- Extended Spread Sheet Database (ESSbase)

Flexfield configurations are preserved across application updates.

# Deployment

The metadata for the flexfield is stored in the application database as soon as you save your configuration changes. Deploying the flexfield generates the ADF business components so that the run time user interface reflects the latest flexfield definition in the metadata.

# Importing and Exporting

Using the Setup and Maintenance work area, you can import and export flexfields across the implementation site. The deployment status must be either Deployed or Deployed to sandbox. Therefore, before you attempt migration, verify and ensure that a flexfield is successfully deployed.

### Run Time

The latest definitions of a flexfield reflect on the user interface at run time only if the flexfield is deployed. When the user interface accesses a business object, the deployed flexfield definition identifies the attributes associated with the captured values. On a page, if you add display customizations for a flexfield using Oracle Composer, the same flexfield segments can appear differently on different pages.

# Patching

Flexfield configurations are preserved during patching and upgrading.

# Flexfields and Value Sets: Highlights

Before you use flexfields to create custom attributes, you should be familiar with the customization layers and the customization life cycle of Oracle Applications Cloud. In addition to the extensive help content available about configuring flexfields, consider the resources below for adding flexfields to business components and alternatives to flexfields where flexfields can't be enabled.

For more information about specific predefined flexfields, open the Setup and Maintenance work area, and use the tasks in the Define Flexfields task list. For customization not available through the tasks and user interface pages, contact My Oracle Support at http://www.oracle.com/pls/topic/lookup?ctx=acc=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc=trs if you are hearing impaired.



Note: Don't use Oracle JDeveloper to customize flexfields.

# Before Configuring Flexfields

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

 For Oracle Sales Cloud, use Application Composer to add custom attributes instead of using descriptive and extensible flexfields.

# Deploying Flexfields

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

See: Customizing SOA Composite Applications

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

# Oracle Business Intelligence

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

See: Enabling Flexfields for Business Intelligence Reporting

See: Importing Changes to Flexfields Automatically

### Related Topics

- Exporting and Moving Customizations: Points to Consider
- Defining Fields: Explained

# Flexfield Management

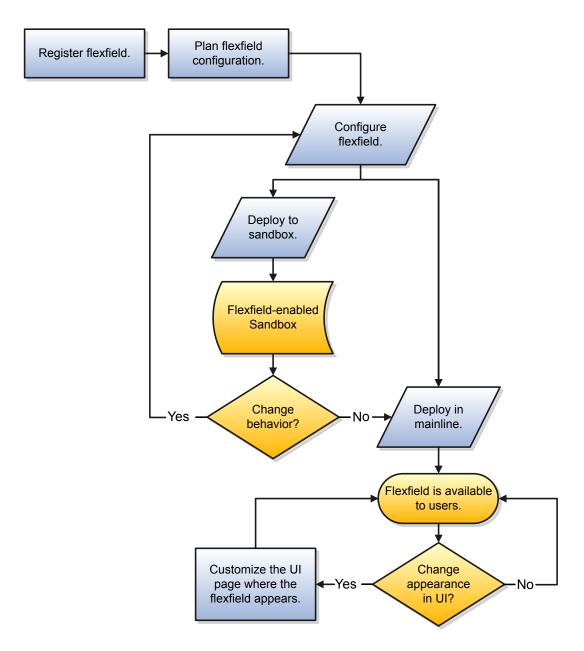
# Managing Flexfields: Points to Consider

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 users. Optionally, you can customize the UI page to change how the flexfield segments appear to users on that page.



The figure shows the processes involved in making flexfields available to users. The tasks in the Define Flexfields activity let administrators configure and deploy flexfields. After you configure and deploy a flexfield to a sandbox, deploy it again to the mainline metadata so that it's available to the users.



Consider the following aspects of managing flexfields:

- Registering flexfields
- Planning flexfields
- Configuring flexfields
- Enabling a flexfields segment for business intelligence
- Deploying flexfields



- · Optionally changing a flexfield segment's appearance in a user interface page
- Identifying flexfields on a run time page and troubleshooting

### Registering Flexfields

A flexfield must be registered before it can be configured. Therefore, application development registers flexfields so that they are available to administrators and implementation consultants for configuration. The registration involves reserving columns of entity tables for use in flexfields. For more information about registering flexfields, see Oracle Fusion Applications Developer's Guide.

### Planning Flexfields

Before you begin planning flexfields, determine what type is appropriate to your needs, and which business objects are available for customizing flexfields. All flexfields consist of segments which represent attributes of an entity. The value a user enters for an attribute is stored in a column of the entity table. Carefully plan flexfields before configuring them. Before configuring new segments for your flexfields, be sure to plan their implementation carefully.

If you have determined that a business object supports flexfields, and those flexfields have been registered, you can begin planning their configuration. Note the code name of the flexfield you intend to configure so that you can find it easily in the Define Flexfield activity. In some cases you can customize how the flexfield appears on the page. See Oracle Applications Cloud Help for specific products to determine any restrictions on using product-specific flexfields.

## 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 run time page.
  - Use the Configure Flexfield icon button to manage all aspects of a flexfield, such as change a segment's sequence number or configure a flexfield segment's business intelligence label.
  - Use the Add Segment and Edit Segment icon buttons to add and edit descriptive or extensible flexfield segments with simple configurations.
  - Use the Add Context icon button to add descriptive or extensible flexfield context values.

### Configuring a flexfield includes the following:

- Defining value sets against which the values entered by users are validated
- Defining the structure or context of the segments in the flexfield
- Specifying the identifying information for each segment
- Specifying the display properties such as prompt, length and data type of each flexfield segment
- · Specifying valid values for each segment, and the meaning of each value within the application
- ▼ Tip: You can create value sets while creating descriptive and extensible flexfield segments. However, define value sets before configuring key flexfield segments that use them, because you assign existing value sets while configuring key flexfield segments.

When creating table-validated, independent, dependent, or subset value sets while creating descriptive and extensible flexfield segments, you can optionally specify to display the description of the selected value to the right of the segment at run time. You can assign sequence order numbers to global segments and to context-sensitive segments in each context. Segment display is always in a fixed order based on the segments' sequence numbers. You cannot enter a number for one segment that is already in use for a different segment. Therefore, you may consider numbering the segments in multiples, such as 4, 5, or 10, to make it easy to insert new attributes.



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

# Enabling a Flexfield Segment for Business Intelligence

You can enable flexfield segments for business intelligence if the flexfield is registered in the database as an Oracle Business Intelligence-enabled flexfield. For more information about enabling segments for business intelligence, see points to consider when enabling descriptive, extensible, and key flexfield segments for business intelligence. For extensible flexfield segments, you can't assign labels to equalize segments across contexts that are semantically equivalent.

### Deploying Flexfields

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

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

In Highlight Flexfields mode, when using the:

- Add Context, Add Segment, and Edit Segment tools for extensible flexfields, use the Save command to save
  your changes. Then use the Deploy command to deploy the flexfield to the mainline metadata
- Add Segment and Edit Segment tools for descriptive flexfields, use the Save and Deploy command to save your changes. Then deploy the flexfield to the mainline metadata

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

### Optionally Changing a Flexfield Segment Appearance

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

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

If the applications are running in different locales, you can provide different translations for translatable text, such as prompts and descriptions. Enter translations using the locale that requires the translated text. In the global area, click your user name and from the **Settings and Actions** menu, select **Set Preferences**. Then change the text to the translated text for that locale.

### Identifying Flexfields on a Run Time Page

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

Even if a descriptive or extensible flexfield isn't yet deployed and no segments appear on the run time page in normal view, the flexfield appears in the Highlight Flexfield view for that page. For descriptive flexfields, the segments as of the last deployment appear. For extensible flexfields, any segments and contexts that have been saved but not yet deployed also appear as disabled.



**Highlight Flexfields** accesses the current flexfield metadata definition. Use the highlighted flexfield's **Configure Flexfield** icon button to manage flexfields directly. Alternatively, note a highlighted flexfield's name to search for it in the tasks for managing flexfields.

For more information about creating flexfields and adding them to a UI page, see the Oracle Fusion Applications Developer's Guide. For more information about customizing flexfield segment appearance with Page Composer, see guidance on customizing existing pages in the Oracle Fusion Applications Extensibility Guide.

# Flexfield Segment Properties: Explained

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

The following aspects are important in understanding

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

### Display Properties

The following table summarizes display properties.

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



Property	Description
	Use instruction help text to provide directions on using the field. If instruction help text is specified, it's appears in an in-field help note window when users move the cursor over the field.

### Properties Related to Search

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

# Range Validation of Segments

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

The conditions for range validation are as follows:

- Segments must be configured for range validation in pairs, one with the low value and one with the high value.
- Both segments must be of the same data type.
- Both segments must be parts of the same structure in a key flexfield or parts of the same context in a descriptive flexfield or extensible flexfield.
- The low value segment must have a sequence number that is lesser than that of 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 detect a low value segment first, and the next range validated segment that it detects must be a high value segment. These two segments are assumed to be a matching pair. The low value and the high value can be equal.

### Rule Validation of Segment Values

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

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

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

#### Naming Conventions

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

The Application Programming Interface (API) name is a name for the segment that isn't exposed to users. The API name is used to identify the segment in various integration points including web services, rules, and business intelligence. Use



alphanumeric characters only with a leading character. For example, enter a code consisting of the characters A-Z, a-z, 0-9 with a non-numeric leading character. The use of spaces, underscores, multi-byte characters, and leading numeric characters isn't permitted. You can't change the API name after the segment has been created.

# Flexfields Segments: How They Are Rendered

Flexfield segments appear on pages as attributes of business objects.

# Settings That Affect Flexfield Segment Display

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

# How Display Type Values Appear

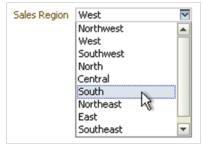
The following figures represent how the display types render on the UI at run time. Each display type screenshot is assigned an alphabet that maps to the display type and its description in the table.

This figure contains the representation of a check box, a drop-down list, a list of values, and a search box.

#### A. Check Box



#### B. Drop-down List



### C. List of Values

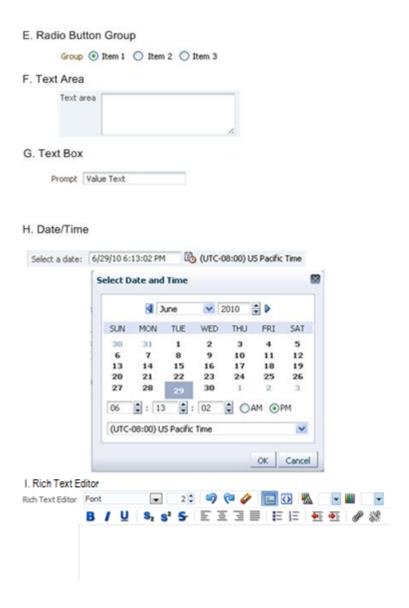


#### D. Pop-up List of Values



This figure contains the representation of a radio button group, text area, text box, date and time, and rich text editor.





This figure contains the representation of a color palette and a static URL field.



# J. Color



### K. Static URL



The following table describes each display type.

Display Type	Figure Reference	Description
Check Box	А	The field appears as a check box. If the user selects the check box, the checked value is used. Otherwise, the deselected value is used.
List	В	The field appears as a list of values available to the user for selection.
List of Values	С	The field appears as a list of values available to the user for selection. The user can also click Search to find more values.
Text Field with Search	D	The field appears as a text field with a Search icon button. The users can type a value in the text field or they can click the Search icon button to open another window for searching.



Display Type	Figure Reference	Description
Radio Button Group	Е	The field appears as a set of radio buttons. The user can select one button. Selecting a button deselects any previously selected button in the set.
Text Area	F	The field appears as a text area in which the user can type multiple lines of text. The display width and height specify the visible width and number of lines in the text area, respectively.
Text Box	G	The field appears as a text field in which the user can type a single line of text. The display width controls the width of the text box.
Date Time	Н	The field enables the user to enter a date if the data type is Date, or a date and time if the data type is Date Time. The user can select the date in a calendar. If the data type is Date Time, the field also displays fields for specifying the hour, minutes, seconds, AM or PM, and time zone.
Rich Text Editor	I	The field appears as a text area in which the user can enter and edit multiple lines of formatted text. The display width and height specify the visible width and number of lines in the rich text editor, respectively.
		Note: This display type is available for extensible flexfields only.
Color	J	The field displays a color palette for the user to select a color at run time and assign it to the segment. During setup, this display type appears in the list for selection only if:
		<ul> <li>You are working on an extensible flexfield segment.</li> <li>The value set for the segment is set to ORA_FND_COLOR_#RRGGBB.</li> </ul>
Static URL	K	The field appears as a text field in which users can enter a fixed URL that opens the web page when clicked.
		Note: The length of the URL must not exceed 255 characters.
		The field isn't displayed.



# Flexfields and Value Sets: How They Work Together

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

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

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

- Defining value sets
- Shared value sets
- Deployment

# Defining Value Sets

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

Note: Ensure that changes to a shared value set are compatible with all flexfield segments that use the value set.

### Shared Value Sets

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

### Value Set Values

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

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

### Deployment

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

# Defaulting and Deriving Segment Values: Explained

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

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



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

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

Initial run time behavior corresponds to the row for the attribute value being created in the entity table. If the default value is read only, it cannot subsequently be changed through the user interface. If the default value isn't read only, users can modify it. However, if the segment value is a derived value, a user-modified segment value is overwritten when the derivation value changes.

Default Type	Default value specified?	Derivation value specified?	Initial run time behavior	Run time behavior after parameter changes
None	No	Yes	No initial segment value	The changed parameter derivation value updates segment value
Constant	Yes	No	Default segment value	N/A
Constant	Yes	Yes	Default segment value	The changed parameter derivation value updates segment value
Parameter	Yes	No	The default segment value is the parameter's default value	N/A
Parameter	Yes	Yes, and same as default value	The default segment value is the parameter's default and derivation value	The changed parameter derivation value updates segment value
Parameter	Yes	Yes, and different from default value	The default segment value is the parameter's default value	The changed parameter default value doesn't update segment value. Only the changed derivation value updates the segment value.

# Flexfield Usages: Explained

The flexfield usage specifies the table with which the flexfield and its segments are associated.

A flexfield can have multiple usages. However, the first table registered for a flexfield indicates the master usage. Segments are based on the master usage. Other usages of the same table for the same flexfield use the same segment setup, though the column names may have a differentiating prefix.

On the Manage Descriptive Flexfields and Manage Extensible Flexfields pages, click the **Show Entity Usages** icon for a specific flexfield to view its entity usage. On the Manage Value Sets page, you can view the flexfield usages for a selected value set.



#### Extensible Flexfields

For extensible flexfield contexts, you can configure a different usage. The usage of an extensible flexfield context determines the scenarios or user interfaces in which the segments of a context appear to end users. For example, the Supplier page displays an extensible flexfield's supplier usage and the Buyer page for the same flexfield displays the buyer usage. Then, a context that is associated only with the supplier usage appears only on the Supplier page and not on the Buyer page.

### Value Sets

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

# FAQs for Flexfield Management

# How can I access predefined flexfields?

Search for predefined flexfields using the Define Flexfields task list:

- 1. In the Setup and Maintenance work area, search for the **Define Flexfields** task list and expand it to view the tasks.
- 2. Open the task that corresponds to the flexfields you are searching for.
- 3. Enter any of the search parameters and click **Search**.
  - Tip: If you don't know the flexfield name or the code, use the Module field to filter search results.
- 4. Click a flexfield to view its details.

# Why can't I edit my flexfield or value set configuration?

Your flexfield or value set configuration may be protected. Application developers mark some configurations as protected, indicating that you can't edit them.

Some examples of configurations that may be protected are:

- Descriptive flexfields
- Extensible flexfield contexts
- Extensible flexfield pages
- Value sets

# Why did my page not display any flexfield?

For a flexfield to be available in the page, it must be registered by developers and also deployed. The segments appear on the page only after you have successfully deployed the flexfield.

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

For information about registering flexfields, see the Oracle Fusion Applications Developer's Guide. Some business objects haven't been designed to support flexfields. For information about how to enable business objects with flexfield capability, see Getting Started with Flexfields in the Oracle Fusion Applications Developer's Guide.

Note: Oracle Sales Cloud doesn't support flexfields.

To add custom attributes to these applications, you may use Application Composer. For more information, see the product-specific documentation.



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

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

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

When you manage Oracle Social Network Objects during setup and maintenance, search for the business object that includes descriptive flexfields. Select the attributes that are defined as flexfield segments and enable them.

# Flexfield Deployment

# Flexfield Deployment: Explained

Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. The deployment process adds custom attributes to the Web Services Description Language (WSDL) schemas exposed by Oracle ADF services and used by SOA composites. Flexfields are deployed for the first time during the application provisioning process. After you configure or change a flexfield, you must deploy it to make the latest definition available to users.

If a descriptive flexfield is enabled for business intelligence, the deployment process redeploys the flexfield's business intelligence artifacts.

You can deploy a flexfield to a sandbox for testing or to the mainline metadata for use in a test or production run time environment. You can deploy extensible flexfields as a background process.

After deployment, the custom attributes are available for incorporating into the SOA infrastructure, such as business process and business rule integration. For example, you can now write business rules that depend on the custom attributes. You must sign out and sign back in to Oracle Applications Cloud to see the changes you deployed in the run time.

The following aspects are important in understanding flexfield deployment:

- Deployment Status
- Initial Deployment Status
- Metadata Validations
- Metadata Synchronization
- Deployment as a Background Process
- Export of Artifacts from Flexfield MDS

### Deployment Status

Every flexfield has a deployment status.

A flexfield can have the following deployment statuses:

Deployment Status	Meaning
Edited	The flexfield metadata definition hasn't been deployed yet. Updates of the metadata definition aren't applied in the run time environment yet.



Deployment Status	Meaning
Patched	The flexfield metadata definition has been modified through a patch or a data migration action, but the flexfield hasn't yet been deployed. So, the updated definition isn't reflected in the run time 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 users. No changes have been made to the flexfield after being deployed to the mainline metadata.
Error	The deployment attempt in the mainline metadata 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 Applications Cloud implementation loads flexfield metadata into the database. This initial load sets the flexfield status to Edited. During installation, the application provisioning process deploys the flexfields of the provisioned applications, setting their status to Deployed if no errors occur.

In a provisioned application, deployed flexfields are ready to use. In some cases, flexfield availability at run time requires setup, such as defining key flexfields.

### Metadata Validation

Use the Validate Metadata command to view possible metadata errors before attempting to deploy the flexfield. Metadata validation is the initial phase of all flexfield deployment commands. By successfully validating metadata before running the deployment commands, you can avoid failures in the metadata validation phase of a deployment attempt. The deployment process ends if an error occurs during the metadata validation phase. Metadata validation results don't affect the deployment status of a flexfield.

#### Metadata Synchronization

When an extensible or descriptive flexfield is deployed, the deployment process regenerates the XML schema definition (XSD). As a result, the custom attributes are available to web services and the SOA infrastructure.

After deploying a flexfield configuration, you must synchronize the updated XML schema definition (XSD) files in the MDS repositories for each SOA application.

Note: To synchronize the updated XSD files in the MDS repositories in Oracle Cloud implementations, log a service request using My Oracle Support at http://support.com/

### Deployment as a Background Process

You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can queue up several extensible flexfields and deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command. When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a



background deployment process. A flexfield's offline status is cleared and its deployment status updated when the background deployment process has completed.

# Export of Artifacts from Flexfield MDS

You can export business components from MDS for descriptive, extensible, or key flexfields, mainly for use in troubleshooting issues with flexfields. Use **Download Flexfield Archive** on the Manage Flexfields page to export MDS artifacts of the selected flexfield, and import them to an archive on your local computer. You can use these archived business components of flexfields for troubleshooting purposes.

Alternatively, export the deployed artifacts using exportMetadata WLST.

# 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 an Oracle Metadata Services (MDS) Repository.

The following aspects are important in understanding how flexfield deployment status is calculated:

- Settings that affect flexfield deployment status
- How deployment status is calculated

# Settings That Affect Flexfield Deployment Status

If you have made a change to a flexfield and expect a changed deployment status, be sure you have saved your changes. No settings affect flexfield deployment status.

# How Deployment Status Is Calculated

If the flexfield definition has been edited through the Define Flexfields activity task flows, the status is Edited. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. Any change, including if a value set used in a flexfield changes, changes the deployment status to Edited. If a flexfield has never been deployed, its status is Edited.

Note: When an application is provisioned, the provisioning framework attempts to deploy all flexfields in that application.

If you deploy the flexfield to a sandbox successfully, the status is Deployed to Sandbox. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a sandbox MDS Repository. Whether the sandbox is active or not doesn't affect the deployment status. If the flexfield was deployed to a sandbox and hasn't been edited or redeployed to the mainline metadata since then, the status remains Deployed to Sandbox independent of whether the sandbox is active, or who is viewing the status.

If you deploy the flexfield successfully to the mainline metadata, 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 metadata.

If either type of deployment fails so that the current flexfield definition isn't deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the Oracle Fusion application likely diverges from the latest successfully deployed flexfield definition.

If the flexfield definition has been modified by a patch, the status is Patched. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. If the flexfield definition was Deployed before the patch and then a patch was applied, the status changes to Patched. If the flexfield definition was Edited before the patch



and then a patch was applied, the status will remain at Edited to reflect that there are still changes (outside of the patch) that aren't yet in effect.

When a deployment attempt fails, you can access the Deployment Error Message for details.

# Deploying a Flexfield-Enabled Sandbox: How It Works With Mainline Metadata

The flexfield definition in a sandbox corresponds to the flexfield metadata definition in the Oracle Fusion Applications database at the time the flexfield was deployed to the sandbox. When the flexfield is ready for end users, the flexfield must be deployed to the mainline metadata.

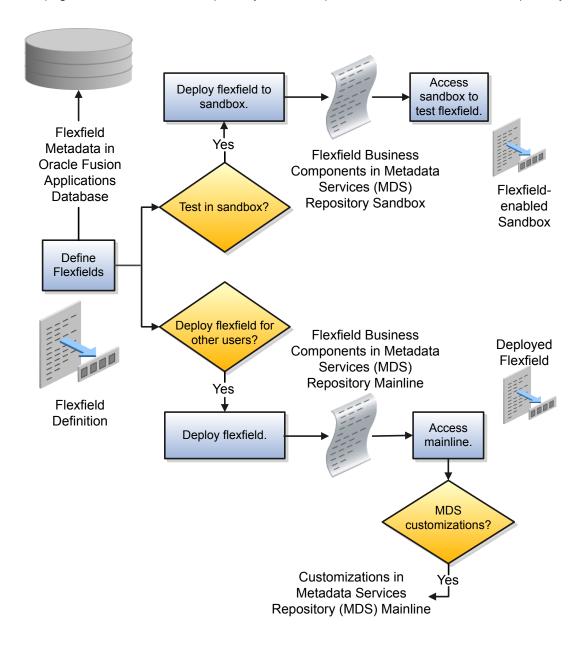
A flexfield-enabled sandbox uses the following components.

- Flexfield metadata in the Oracle Applications Cloud database
- Flexfield business components in a sandbox Oracle 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 metadata applies the flexfield definition to the mainline MDS Repository where it is available to end users. After



deploying the flexfield to the mainline metadata, 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.

Caution: Don't customize flexfield segment display properties using Page Composer in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.



# Mainline Metadata Services Repository Data

The Oracle Fusion Applications database stores the single source of truth about a flexfield. When the flexfield is deployed, the ADF business component objects that implement the flexfield in the run time user interface are generated in the mainline MDS Repository from this source.

#### Related Topics

• Managing Customizations Using Sandboxes: Explained

# Deploying a Flexfield to a Sandbox: Points to Consider

Deploying a flexfield to a sandbox creates a flexfield-enabled sandbox. Each flexfield-enabled sandbox contains only one flexfield.

You can test the run time behavior of a flexfield in the flexfield-enabled sandbox. If changes are needed, you return to the Define Flexfield tasks to change the flexfield definition.

When you deploy a flexfield to sandbox, the process reads the metadata about the segments from the database, generates flexfield Application Development Framework (ADF) business component artifacts based on that definition, and stores in the sandbox only the generated artifacts derived from the definition.

When you deploy a flexfield sandbox, the process generates the name of the flexfield sandbox, and that flexfield sandbox is set as your current active sandbox. When you next sign in to the application, you can see the updated flexfield configurations. The Oracle Fusion Applications global area displays your current session sandbox.

Note: Unlike a standalone sandbox created using the Manage Sandboxes tool, the sandbox deployed for a flexfield contains only the single flexfield. You can manage flexfield sandboxes, such as setting an existing flexfield sandbox as active or deleting it, using the Manage Sandboxes tool.

When you deploy a flexfield to the mainline metadata after having deployed it to the sandbox, the sandbox-enabled flexfield is automatically deleted.

### Sandbox MDS Repository Data

The sandbox data lets you test the flexfield in isolation without first deploying it in the mainline metadata where it could be accessed by users.

Caution: Don't customize flexfield segment display properties using Page Composer in a flexfield-enabled

sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.

Managing a Flexfield-Enabled Sandbox

When you deploy a flexfield as a sandbox, that flexfield-enabled sandbox automatically gets activated in your user session. When you sign back in to see the changes, the sandbox is active in your session.

You can only deploy a flexfield to a sandbox using the Define Flexfields task flow pages.

You also can use the Manage Sandboxes feature in the Administration menu of the Setup and Maintenance work area to activate and access a flexfield-enabled sandbox.

Note: Whether you use the Define Flexfields or Manage Sandboxes task flows to access a flexfield-enabled sandbox, you must sign out and sign back in before you can see the changes you deployed in the run time.



You cannot publish the flexfield from the sandbox to the mainline metadata. You must use the Define Flexfields task flow pages to deploy the flexfield for access by users of the mainline metadata because the flexfield configuration in the mainline metadata is the single source of truth.

#### Related Topics

Managing Customizations Using Sandboxes: Explained

# 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 Oracle Metadata Services (MDS) Repository with predefined flexfield artifacts and for deploying flexfields.

The table describes the available commands.

WebLogic Server Tool Command	Description
deployFlexForApp	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.
deployFlex	Deploy a single flexfield regardless of deployment status
deployPatchedFlex	Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.
deleteFlexPatchingLabels	Displays MDS label of flexfield changes for viewing and deleting patching labels.
validateFlexDeploymentStatus	Displays list containing flexfields that aren't 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 errors occurred. If a run time exception occurs, the output displays the trace back 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
- Closing 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 Oracle Fusion Middleware Extensions for Oracle Application.

For more information about deploying the Oracle Fusion Middleware Extensions for Oracle Application to the server domains, see the Oracle Fusion Applications Developer's Guide.

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

Start the WebLogic Server Tool (WLST) if 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 about 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.

- Note: This command is run automatically when you provision applications. However, if you customize applications, you have to manually run it following the order of tasks as given here:
  - 1. Configure your application to read the flexfield artifacts from the MDS Repository.
  - 2. Run the deployFlexForApp command.
  - 3. Sign in to the application.

This sequence of steps is required even if there is no predefined flexfield metadata.

This command doesn't deploy flexfields that have a status of Deployed unless the force parameter is set to 'true' (the default setting is 'false').

For more information about priming the MDS partition with configured flexfield artifacts, see the Oracle Fusion Applications Developer's Guide.

From the WLST tool, execute the following commands to deploy the artifacts to the MDS partition, replacing product\_application\_shortname with the application's short name wrapped in single-quotes.

```
deployFlexForApp('product_application_shortname'[, 'enterprise_id'] [,'force'])
```



In a multi-tenant environment, replace <code>enterprise\_id</code> with the Enterprise ID to which the flexfield is mapped. Otherwise, replace with <code>'None'</code> or don't provide a second argument.

To deploy all flexfields regardless of their deployment status, set force to 'true' (the default setting is 'false'). To deploy all flexfields in a single-tenant environment, you either can set enterprise\_id to 'None', or you can use the following signature:

```
deployFlexForApp(applicationShortName='product_application_shortname',force='true')
```

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, either descriptive flexfield, key flexfield, or extensible flexfield. The values must be wrapped in single-quotes.

```
deployFlex('flex_code', 'flex_type')
```

Optionally, execute the following command if the flexfield is an extensible flexfield, and you want to deploy all the flexfield's configurations.

Note: By default, extensible flexfields are partially deployed. That is, only the pages, contexts, or categories that had recent changes, are deployed.

```
deployFlex('flex_code', 'flex_type', ['force_Complete_EFF_Deployment'])
where, forceCompleteEFFDeployment=None
```

### Using the deployPatchedFlex Command

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

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

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

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

```
deployPatchedFlex()
```

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

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

# Using the deleteFlexPatchingLabels Command

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

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

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

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



# Using the validateFlexDeploymentStatus Command

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

#### validateFlexDeploymentStatus()

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

### Closing WLST and Checking the Results

To close the tool, execute the command: disconnect().

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

# Manage Value Sets

# Value Sets: Explained

A value set is a group of valid values that you assign to a flexfield segment to control the values that are stored for business object attributes.

An end user enters a value for an attribute of a business object while using the application. The flexfield validates the value against the set of valid values that you configured as a value set and assigned to the segment.

For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

Note: Ensure that changes to a shared value set are compatible with all flexfields segments using the value set.

The following aspects are important in understanding value sets:

- Managing value sets
- Validation
- Security
- Precision and scale
- Usage and deployment
- Protected value set data

### Managing Value Sets

To access the Manage Value Sets page, use the Manage Value Sets task, or use the Manage Descriptive Flexfields and Manage Extensible Flexfields tasks for configuring a segment, including its value set. To access the Manage Values page, select the value set from the Manage Value Sets page, and click **Manage Values**. Alternatively, click **Manage Values** from the Edit Value Set page.



#### Validation

The following types of validation are available for value sets:

- · Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set doesn't present a list of valid values to users. Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

Note: For the Accounting Key Flexfield value sets, you must use independent validation only. If you use other validations, you can't use the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration.

# Security

Value set security only works in conjunction with usage within flexfield segments. You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.

The application of value set security has the following conditions:

- At the value set level: The value set is the resource secured by data security policies. If a value set is secured, every usage of it in any flexfield is secured. It isn't possible to disable security for individual usages of the same value set.
- Applies to independent, dependent, or table-validated value sets.
- Applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes. Value set security doesn't determine which descriptive flexfield data is shown upon querying.
- Security conditions defined on value sets always use table aliases. When filters are used, table aliases are always used by default. When predicates are defined for data security conditions, make sure that the predicates also use table aliases.

For key flexfields, the attributes in the view object that correspond to the code combination ID (CCID), structure instance number (SIN), and data set number (DSN) cannot be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the CCID segment is the common attribute.

#### Precision and Scale

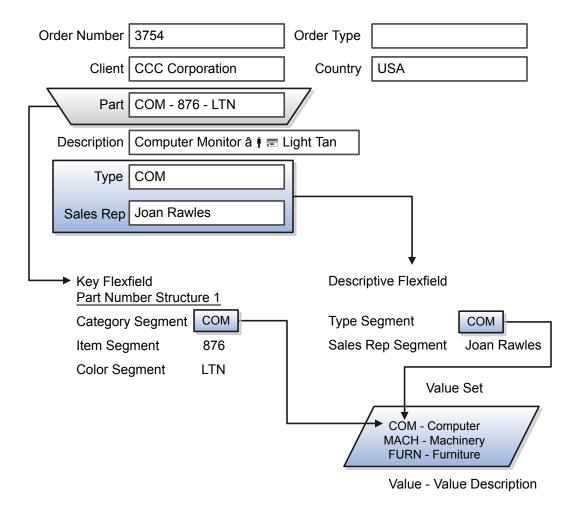
If the data type of a value set is Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

### Usage and Deployment

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.



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



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

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

### Protected Value Set Data

Application developers may mark some value sets as protected, indicating that you can't edit them.

You can edit only value sets that are not marked as protected. You can't edit or delete protected value sets. If the value set type supports values (such as independent, dependent or subset value sets), then you can't add, edit, or delete values.

Note: There is no restriction on references to protected value sets. Value sets, protected or not, may be assigned to any flexfield segment. Likewise, other value sets may reference protected value sets; for example, an unprotected dependent value set may reference a protected independent value set.



#### Related Topics

• Chart of Accounts: How Its Components Fit Together

# Defining Value Sets: Critical Choices

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

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

The following aspects are important in defining value sets:

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

# Value Sets for Context Segments

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

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

### Format Only Validation

The format only validation type enables users to enter any value, as long as it meets your specified formatting rules. 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 permits only numeric characters, users can enter the value 456 (for a value set with maximum length of three or more), but can't enter the value ABC. A format only value set doesn't otherwise restrict the range of different values that users can enter. For numeric values, you can also specify if a numeric value should be zero filled or how may digits should follow the radix separator.

# Interdependent Value Sets

Use an independent value set to validate data against a list that isn't stored in an application table, and not dependent on a subset of another independent value set. You cannot specify a dependent value set for a given segment without having first defined an independent value set that you apply to another segment in the same flexfield. Use a dependent value set to limit the list of values for a given segment based on the value that the user has defined 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 the states in the USA with values such as CA, NY, and so on. Then you define a dependent value set of cities in the USA with values such as San Francisco and Los Angeles that are valid



for the independent value CA. Similarly, New York City and Albany 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. Users don't have to select a value for another segment first to have access to the subset value set.

Independent, dependent, and subset value sets require a customized list of valid values. Use the Manage Values page to create and manage a value set's valid values and the order in which they appear.

💡 Tip: You can customize the Manage Value Sets page to capture additional attributes for each valid value by adding context-sensitive segments in a new context for FND\_VS\_VALUES\_B descriptive field.

#### **Table Validation**

Typically, you use a table-validated set when the values you want to use are already maintained in an application table, such as a table of supplier names. Specify the table column that contains the valid value. You can optionally specify the description and ID columns, a WHERE clause to limit the values to use for your set, and an ORDER BY clause.

If you specify an ID column, then the flexfield saves the ID value, instead of the value from the value column, in the associated flexfield segment. If the underlying table supports translations, you can enable the display of translated text by basing the value set's value column on a translated attribute of the underlying table. You should also define an ID column that is based on an attribute that isn't language-dependent so that the value's invariant ID (an ID that doesn't change) is saved in the transaction table. The run time displays the corresponding translated text from the value column for the run time session's locale.

Table validation lets you enable a segment to depend upon multiple prior segments in the same context structure. You cannot reference other flexfield segments in the table-validated value set's WHERE clause. That is, the WHERE clause cannot reference SEGMENT.segment code or VALUESET.value set code.

Table-validated value sets have unique values across the table, irrespective of bind variables. The WHERE clause fragment of the value set is considered if it doesn't have bind variables. If it has bind variables, the assumption is that the values are unique in the value set. If you use table validated value sets for key flexfields, then you can't use all integration functionalities supported for key flexfields, such as:

- Data security
- Oracle Transactional Business Intelligence (OTBI)
- Extended Spread Sheet Database (ESSbase)
- Tree or hierarchy integration

To use these integration functionalities for key flexfields, you must use independent value sets only.

### Range

In the case of format, independent, or dependent value sets, you can specify a range to limit which values are valid. You can specify a range of values that are valid within a value set. You can also specify a range validated pair of segments where one segment represents the low end of the range and another segment represents the high end of the range.

For example, you might specify a range for a format-only value set with format type Number where the user can enter only values between 0 and 100.

### Security

In the case of independent and dependent values, you can specify that data security be applied to the values in segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment users can view or modify.



To enable security on a value set, specify a database resource, typically the code value for the value set. Using the Manage Database Security Policies task, specify conditions, such as filters or SQL predicates, and policies that associate roles with conditions. You can use a filter for simple conditions. For more complex conditions, use a SQL predicate.

Value set data security policies and conditions differ from data security conditions and policies for business objects in the following ways:

- You can grant only read access to users. You cannot specify any other action.
- When defining a condition that is based on a SQL predicate, use VALUE, VALUE\_NUMBER, VALUE\_DATE, VALUE\_TIMESTAMP, or VALUE\_ID to reference the value from a dependent, independent, or subset value set. For table value sets, use a table alias to define the table, such as &TABLE\_ALIAS category=70.

When you enable security on table-validated value sets, the security rule that is defined is absolute and not contingent upon the bind variables (if any) that may be used by the WHERE clause of the value set. For example, suppose a table-validated value set has a bind variable to further filter the value list to x, y and z from a list of x, y, z, xx, yy, zz. The data security rule or filter written against the value set must not assume anything about the bind variables. Instead the whole list of values must be available and you write the rule, for example, to permit x, or to permit y and z. By default in data security, all values are denied and show only rows to which access has been provided.

#### Testing and Maintenance

You don't have to define or maintain values for a table-validated value set, as the values are managed as part of the referenced table or independent value set, respectively.

You cannot manage value sets in a sandbox.

When you change an existing value set, the deployment status for all affected flexfields changes to Edited. You must redeploy all flexfields that use that value set to make the flexfields reflect the changes. In the UI pages for managing value sets, the value set's usages show which flexfields are affected by the value set changes.

If your application has more than one language installed, or there is any possibility that you might install one or more additional languages for your application in the future, select **Translatable**. This doesn't require you to provide translated values now, but you cannot change this option if you decide to provide them later.

## Planning Value Sets: Points to Consider

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

The following aspects are important in planning value sets:

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

#### List of Values

You can use one of the following types of lists to specify the valid values for a segment:

- Table column
- Custom list. Also include a sub list.
- Dependent custom list



If the valid values exist in a table column, use a table value set to specify the list of values. To limit the valid values to a subset of the values in the table, use a SQL WHERE clause. Table value sets also provide some advanced features, such as enabling validation depending on other segments in the same structure.

Use an independent value set to specify a custom set of valid values. For example, you can use an independent value set of Mon, Tue, Wed, and so forth to validate the day of the week. You can also specify a subset of an existing independent value set as the valid values for a segment. For example, if you have an independent value set for the days of the week, then a weekend subset can be composed of entries for Saturday and Sunday.

Use a dependent value set when the available values in the list and the meaning of a given value depend on which independent value was selected for a previously selected segment value. For example, the valid holidays depend on which country you are in. A dependent value set is a collection of value subsets, with one subset for each value in a corresponding independent value set.

For lists of values type value sets, you can additionally limit the valid values that an end user can select or enter by specifying format, minimum value, and maximum value. For list of values type value sets, you can optionally implement value set data security. If the Oracle Fusion applications are running in different locales, you might need to provide different translations for the values and descriptions.

### Plain Text Input

Use a format-only value set when you want to allow end users to enter any value, as long as that value conforms to formatting rules. For example, if you specify a maximum length of 3 and numeric-only, then end users can enter 456, but not 4567 or 45A. You can also specify the minimum and maximum values, whether to right-justify, and whether to zero-fill. With a format-only value set, no other types of validation are applied.

## Value Ranges

You can use either a format-only, independent, or dependent value set to specify a range of values. For example, you might create a format-only value set with Number as the format type where the end user can enter only the values between 0 and 100. Or, you might create a format-only value set with Date as the format type where the end user can enter only dates for a specific year, such as a range of 01-JAN-93 to 31-DEC-93. Because the minimum and maximum values enforce these limits, you need not define a value set that contains each of these individual numbers or dates.

### Value Format

Flexfield segments commonly require some kind of format specification, regardless of validation type. Before creating a value set, consider how you will specify the required format.

The following table shows options for validation type and value data type.

Option	Description
Value data type	Character, Number, Date, Date Time.
Value subtype	Text, Translated text, Numeric digits only, Time (20:08), Time (20:08:08).
	An additional data type specification for the Character data type for the Dependent, Independent, and Format validation types.
Maximum length	Maximum number of characters or digits for Character data type.
Precision	Maximum number of digits the user can enter.
Scale	Maximum number of digits that can follow the decimal point.



Option	Description	
Uppercase only	Lowercase characters automatically changed to uppercase.	
Zero fill	Automatic right-justification and zero-filling of entered numbers (affects values that include only the digits 0-9).	

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

#### Value Sets for Context Segments

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

### Security

When enabling security on a value set, the data security resource name is an existing value set or one that you want to create. The name typically matches the code value for the value set. You cannot edit the data security resource name after you save your changes.

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

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

The following bind variables refer to flexfield elements:

- :{SEGMENT.<segment code>}
- :{CONTEXT.<context\_code>;SEGMENT.<segment\_code>}
- :{VALUESET.<value\_set\_code>}
- :{FLEXFIELD.<internal code>}
- :{PARAMETER.<parameter\_code>}

#### Segment Code

#### : {SEGMENT.<segment\_code>}

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

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

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



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

#### Context Code

### : {CONTEXT.<context\_code>;SEGMENT.<segment\_code>}

This bind variable, which is valid only for extensible flexfields, refers to the ID (if the value set is ID-validated) or value (if not ID-validated) of a segment that is in a different context than the target segment (the segment with the WHERE clause).

- The <context\_code> identifies the context and must be in the same category or in an ancestor category. It cannot be a multiple-row context.
- The <segment\_code> identifies the segment. The data type of the bind value is the same as the data type of the segment's column.
- Note: The target segment should appear in the UI after the source segment to ensure the source segment has a value. If the target segment's context is a single-row context, the source and target segments must be on separate pages and the target page must follow the source page.

The framework of extensible flexfields doesn't perform any additional validation related to mismatched values for segments defined with cross context bind parameters. Administrators must populate the correct pair of segment values.

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

#### Value Set Code

```
:{VALUESET.<value_set_code>}
```

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

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

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

#### Flexfield Internal Code

#### :{FLEXFIELD.<internal code>}

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

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



- CONTEXT\_CODE the context code of the flexfield context in which this value set is used. The data type of CONTEXT\_CODE and its resulting bind value is VARCHAR2.
- SEGMENT\_CODE the identifying code of the flexfield segment in which this value set is used. The data type of SEGMENT\_CODE and its resulting bind value is VARCHAR2.
- VALIDATION\_DATE the current database date. The data type of VALIDATION\_DATE and its resulting bind value is DATE.

#### Flexfield Parameters

#### :{PARAMETER.<parameter\_code>}

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

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

## Table-Validated Value Set: Worked Example

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

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

Property	Value	
FROM clause	FND_LOOKUPS	
WHERE clause	lookup_type = 'CN_ XX_ CUST_ SATISFACT_ SCORE'	
ID column	lookup_code	
Value column	meaning	
Description column	description	
Enable Flag column	enabled_flag	
Start Date column	start_ date_active	
End Date column	end_ date_active	
Order by	display_ sequence	



After completing this task, you should have created your customer satisfaction value set for the Incentive Compensation page of your implementation project.

## Creating a Value Set Based on a Lookup

- 1. From the Setup and Maintenance work area, find the Manage Value Sets task and click the **Go to Task** icon button.
- 2. On the Manage Value Sets page, click the **Create** icon button.
- 3. On the Create Value Set page, enter the following values:
  - a. In the Value Set Code field, enter CN XX CUSTOMER SATISFACTION SCORES
  - **b.** In the Description field, enter Customer satisfaction score.
  - c. In the Module field, select Search....
  - d. In the Search and Select: Module subwindow, enter Incent in the User Module Name field
  - e. Select Incentive Compensation.
  - f. Click OK.
- 4. On the Create Value Set page, enter the following values:
  - a. In the Validation Type field, select Table.
  - **b.** In the Value Data Type field, select Character.
  - c. In the Definition section FROM Clause field, enter FND\_LOOKUPS.
  - d. In the Value Column Name field, enter DESCRIPTION.
  - e. In the Description Column Name field, enter MEANING.
  - f. In the ID Column Name field, enter LOOKUP\_CODE.
  - g. In the Enabled Flag Column Name field, enter 'Y'.
  - h. In the Start Date Column Name field, enter START DATE ACTIVE.
  - i. In the End Date Column Name field, enter END DATE ACTIVE.
  - j. In the WHERE Clause field, enter LOOKUP TYPE = 'CN XX CUST SATISFACT SCORE'.
- 5. Click Save and Close.
- 6. In the Manage Value Sets page, click Done.

## Adding Attributes to the Manage Value Sets Page: Procedures

You can add attributes to independent, dependent, and subset value sets. The attributes appear on the Manage Value Sets page where you can store additional information about each valid value. To display attributes on an application page, you must programmatically modify the application.

To add attributes and subsequently view them on the Manage Value Sets page, perform the following steps:

- 1. Using the Manage Descriptive Flexfields task, find the FND\_VS\_VALUES\_B flexfield and open it for editing.
- 2. Click Manage Contexts.
- 3. Create a new context and use the value set code for the context code.
- **4.** Add new attributes as context-sensitive segments and save the changes.
- 5. Deploy FND\_VS\_VALUES\_B to run time.
- 6. Sign out and sign back in.
- 7. Open the Manage Value Sets page to view the new attributes.

## Importing Value Set Values: Procedure

You can import a file containing values that you want to edit or add to a given independent or dependent value set.

For example, uploading a hundred values may be more efficient than creating them individually using the Manage Value Sets task. However, for just a few values, it may be quicker to perform the relevant tasks.



## Importing Value Set Values

To import value set values:

1. Create a flat file containing the values in the value set that you want to add or update.

#### Note:

- When creating the file, you must specify an existing value set code to which you want to add values
  or edit existing values. If the value set does not exist, add the value set using the appropriate Manage
  Value Sets setup task in the Setup and Maintenance work area.
- The file that you create must adhere to the formatting and content requirements for creating flat files containing value set values.
- 2. Upload the flat file to the content repository using the Files for Import and Export page.
- 3. Import the file using the appropriate Manage Value Sets setup task in the Setup and Maintenance work area. To import the file:
  - a. Click Actions Import in the Manage Value Sets page.
  - b. In the File Name field, enter the name of the flat file you uploaded using the Files for Import and Export page.
  - c. In the Account field, select the user account containing the flat file.
  - d. Click Upload.
  - Note: Alternatively, you can import the file using either of the following methods:
    - o Run the Upload Value Set Values scheduled process.
    - Use the Applications Core Metadata Import web service. For more information on the Applications Core Metadata Import web service, see the SOAP Web Services guide for your cloud services.

#### Related Topics

• Files for Import and Export: Explained

## Requirements for Flat Files to Upload Value Set Values: Explained

You can import large volumes of value set value data from the content repository. To upload value set values to the content repository, create a flat file containing the values in the value set that you want to add or update. You upload these flat files to the content repository using the Files for Import and Export page.

#### General Requirements

The first line of the flat file must contain the column names for the value set value data, including all mandatory columns, and separated by the '|' (pipe) character. Each subsequent line should contain a row of data specified in the same order as the column names, also separated by the '|' character.

The requirements for creating flat files vary with the type of value sets:

- Independent value sets
- Dependent value sets

## Independent Value Set

A flat file for uploading values for independent value sets must contain the following mandatory columns:



Column Name	Data Type
ValueSetCode	VARCHAR2(60)
Value	VARCHAR2(150)
Enabled Flag	VARCHAR2(1), Y or N

Note: You can also specify optional columns.

#### Examples:

• To upload values to a COLORS independent value set with the minimum columns, you can use the following flat file:

```
ValueSetCode | Value | EnabledFlag
COLORS | Red | Y
COLORS | Orange | Y
COLORS | Yellow | Y
```

• To upload values to a STATES independent value set with more (optional) columns, you can use the following flat file:

```
ValueSetCode | Value | Description | EnabledFlag
STATES | AK | Alaska | Y
STATES | CA | California | Y
STATES | WA | Washington | Y
```

## Dependent Value Sets

A flat file for uploading values for dependent value sets must contain the following mandatory columns:

Column Name	Data Type
Value Set Code	VARCHAR2(60)
Independent Value	VARCHAR2(150)
Value	VARCHAR2(150)
Enabled Flag	VARCHAR2(1), Y or N

Note: You can also specify optional columns.

#### Example:

To upload values to a CITIES dependent value set (dependent on the STATES independent value set), you can use the following flat file:

```
ValueSetCode | IndependentValue | Value | EnabledFlag
CITIES | AK | Juneau | Y
CITIES | AK | Anchorage | Y
CITIES | CA | San Francisco | Y
```



```
CITIES | CA | Sacramento | Y
CITIES | CA | Los Angeles | Y
CITIES | CA | Oakland | Y
```

## Additional Optional Columns

In addition to the mandatory columns, you can add the following optional columns for both dependent and independent value sets:

Туре
VARCHAR2(150), for use in value sets that are translatable
VARCHAR2(240)
DATE, formatted as YYYY-MM-DD
DATE, formatted as YYYY-MM-DD
NUMBER(18)
VARCHAR2(30)
VARCHAR2(30)
VARCHAR2(30)

#### Related Topics

Files for Import and Export: Explained

## Upload Value Set Values Process

This process uploads a flat file containing value set values for flexfields. You can use this scheduled process to upload a file containing values you want to edit or add to an existing independent or dependent value set. This process is useful for adding or updating large volumes of value set value data in an automated or recurring fashion. For example, you can upload a hundred values on a recurring basis when scheduled as a recurring process. This method may be more efficient than using the one-time Import action in the Manage Value Sets tasks in the Setup and Maintenance work area. However, for an ad hoc task of uploading a hundred values, it may be quicker to use the Import action in the relevant tasks.

Run this process from the Scheduled Processes Overview page. You can run it on a recurring basis whenever the flat file in the content repository account is updated.

You must create the flat file containing the values data, and upload the flat file to the content repository using the Files for Import and Export page.

#### **Parameters**

#### **Flat File Name**



Enter the name of the flat file you uploaded using the Files for Import and Export page.

#### Account

Select the user account containing the flat file in the content repository to upload.

### Related Topics

• Files for Import and Export: Explained

• Scheduled Processes: Explained

## Translating Flexfield and Value Set Configurations: Explained

When you first configure a flexfield or segment, the translatable text that you enter, such as prompts and descriptions, is stored as the text for all installed locales. You may then provide a translation for a given locale, then the value that was first entered is used for that locale.

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

After you complete the translations, deploy the flexfield.

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

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

# FAQs for Manage Value Sets

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

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

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

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



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

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

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

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

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

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

# Manage Descriptive Flexfields

## Descriptive Flexfields: Explained

Use descriptive flexfields to add custom attributes to business object entities, and define validation for them.

All the business object entities that you can use in the application are enabled for descriptive flexfields. However, configuring descriptive flexfields is an optional task.

### Context

A descriptive flexfield can have only one context segment to provide context sensitivity. The same underlying database column can be used by different segments in different contexts.

For example, you can define a Dimensions context that uses the following attributes:

- ATTRIBUTE1 column for height
- ATTRIBUTE2 column for width
- ATTRIBUTE3 column for depth

You can also define a Measurements context that uses the same columns for other attributes:

- ATTRIBUTE1 column for weight
- ATTRIBUTE2 column for volume
- ATTRIBUTE3 column for density

### Segments and Contexts

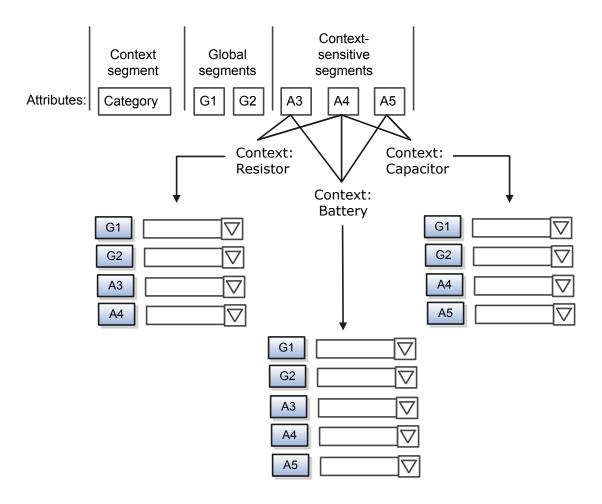
Descriptive flexfield segments are of the following types:

Segment Type	Run Time Appearance
Global segment	Always available
Context segment	Determines which context-sensitive segments are displayed



Segment Type	Run Time Appearance
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. Additionally, the descriptive flexfield comprises two global segments that appear in each context, and three context-sensitive segments that only appear in the specific context.



Application development determines the number of segments available for configuring. During implementation, configure the flexfield by determining the following:

- Attributes to add using the available segments
- Context values
- The combination of attributes in each context

#### Value Sets

For each global and context-sensitive segment, you configure the values permitted for the segment. Based on it, the values that end users enter are validated, including interdependent validation among the segments.



#### Protected Descriptive Flexfield Data

Application developers may mark some data configurations in a descriptive flexfield as protected, indicating that you can't edit them.

## Planning Descriptive Flexfields: Points to Consider

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users. Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list, Manage Sandboxes, and Highlight Flexfields for adding and editing flexfield segments.

Planning a descriptive flexfield can involve the following tasks:

- 1. Identify existing parameters.
- 2. Identify existing context values and whether the context value is derived.
- 3. Identify custom attributes and plan the descriptive flexfield segments, segment properties, and structure.
- 4. Plan validation rules.
- 5. Plan initial values.
- 6. Plan attribute mapping to Oracle Business Intelligence objects.

#### Identify Existing Descriptive Flexfield Parameters

Some descriptive flexfields provide parameters that can be used to specify the initial value of a descriptive flexfield segment. The parameter is external reference data, such as a column value or a session variable. For example, if a flexfield has a user email parameter, you can configure the initial value for a customer email attribute to be derived from that parameter.

Review the list of available parameters in the **Derivation Value** field in the Create Segment page for a descriptive flexfield. If you decide to use one of the parameters to set an initial value, select that parameter from the **Derivation Value** drop-down list when you add the descriptive flexfield segment.

#### Evaluate Whether the Context Value Is Derived

The context value for a descriptive flexfield might have been preconfigured to be derived from an external reference. For example, if the context is Marriage Status, then the value might be derived from an attribute in the employee business object. When the context value is derived, you might need to take the derived values and their source into consideration in your plan.

To determine whether the context value is derived, access the Edit Descriptive Flexfield task to view the list of configured context values for the flexfield. The Derivation Value field in the Context Segment region displays a list of available parameters. If context values have been preconfigured, see Oracle Applications Cloud Help for product-specific information about the use of those values.

#### Plan the Segments, Segment Properties, and Structure

Identify the custom attributes you need for a business object to determine the segments of the descriptive flexfield. Determine the segment properties such as the prompt, display type, or initial value.

The structure of the descriptive flexfield is determined by its global, context, and context-sensitive segments. Plan a global segment that captures an attribute for every instance of the business object. Plan a context for segments that depend on a condition of situation applying to a particular instance of the business object. Plan context-sensitive segments to capture attributes that are relevant in the context.

There is only one context segment available for descriptive flexfields. If you have more than one group of custom attributes where you could use the context segment, you will have to pick one group over the others, based on your company's needs and priorities, and add the other custom attributes as global segments.



#### Plan Validation Rules

Define each segment's validation rules and check if value sets exist for those rules or you must create new ones. If you must create a value set, you can create it either before configuring the flexfield or while creating or editing a segment.

When determining a segment's validation rules, consider the following questions:

- What is the data type character, date, date and time, or number?
- Does the segment require any validation beyond data type and maximum length?
- Should a character type value be restricted to digits, or are alphabetic characters allowed?
- Should alphabetic characters automatically be changed to uppercase?
- Should numeric values be zero-filled?
- How many digits can follow the radix separator of a numeric value? In base ten numerical systems the radix separator is decimal point.
- Does the value need to fall within a range?
- Should the value be selected from a list of valid values? If so, consider the following questions:
  - Can you use an existing application table from which to obtain the list of valid values, or do you need to create a custom list?
  - o If you are using an existing table, do you need to limit the list of values using a WHERE clause?
  - o Does the list of valid values depend on the value in another flexfield segment?
  - o Is the list of valid values a subset of another flexfield segment's list of values?

#### Plan Initial Values

For every segment, list the constant value or SQL statement, if any, to use for the initial value of the custom attribute.

#### Plan How Segments Map to Oracle Business Intelligence Objects

You can extend descriptive flexfields into Oracle Transactional Business Intelligence (OTBI) for ad hoc reporting purposes. Determine the descriptive flexfield segments to be made available for reporting, and select the **BI Enabled** check box accordingly on the Manage Descriptive Flexfields page. You must run a process to extend the BI enabled segments into OTBI. For more information about extending the BI enabled segments into OTBI, see the Setup and Configuration chapter in the Oracle Transactional Business Intelligence Administrator's Guide.

Depending on the reporting needs, you may map similar context-sensitive attributes from different contexts to the same attribute in OTBI. For example, there may be a segment tracking the Product Color attribute in different contexts of a context sensitive descriptive flexfield. You can use segment labels to map these context-sensitive attributes together by defining a segment label and updating the BI Label list accordingly.

## Managing Descriptive Flexfields: Points to Consider

Configuring descriptive flexfields involves managing the available flexfields registered with your Oracle Applications Cloud database and configuring their flexfield-level properties, defining and managing descriptive flexfield contexts, and configuring global and context-sensitive segments.

Every descriptive flexfield is registered to include a context segment, which you may choose to use or not.

In general, configuring descriptive flexfields involves:

- 1. Creating segment labels for business intelligence enabled flexfields.
- 2. Configuring global segments by providing identity information, the initial default value, and the display properties.
- 3. Configuring the context segment by specifying the prompt, whether the context segment should be displayed, and whether a value is required.



**4.** Configuring contexts by specifying a context code, description, and name for each context value, and adding its context-sensitive segments, each of which is configured to include identifying information, the column assignment, the initial default value, and the display properties.

The following aspects are important in understanding descriptive flexfield management:

- Segments
- Adding segments to highlighted descriptive flexfields
- Usages
- Parameters
- Delimiters
- Initial Values
- Business Intelligence

### Segments

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

Value sets are optional for context segments and follow specific guidelines:

- The value set that you specify for a context segment consists of a set of context codes.
- Each context code corresponds to a context that is appropriate for the descriptive flexfield.
- The value set must be independent or table-validated.
- If table-validated, the WHERE clause must not use the VALUESET.value\_set\_code or SEGMENT.segment\_code bind variables.
- The value set must be of data type Character with the maximum length of values being stored no larger than the context's column length.
- If you don't specify a value set for a context segment, the valid values for that context segment are derived from the
  context codes. The definition of each context segment specifies the set of context-sensitive segments that can be
  presented when that context code is selected by the end user.
- For reasons of data integrity, you cannot delete an existing context. Instead, you can disable the associated context value in its own value set by setting its end date to a date in the past.
- You can configure the individual global segments and context-sensitive segments in a descriptive flexfield. These segment types are differentiated by their usage, but they are configured on application pages that use most of the same properties.

#### Adding Segments to Highlighted Descriptive Flexfields

When you highlight flexfields on a run time page and use an **Add Segment** icon button to create a segment, the segment code, name, description, table column, and sequence number are set automatically. If you use an **Add Segment** icon button to configure descriptive flexfield segments, you cannot use an existing value set. Value sets are created automatically when you add the segments. You can enter the valid values, their descriptions, and the default value or specify the formatting constraints for the value set, such as minimum and maximum values.

Depending on display type, the value set you create with the **Add Segment** icon button is either an independent value set or a format-only value set. The table shows which type of value set is created depending on the segment display component you select.

Display Component	Value Set Created with Add Segment
Check Box	Independent



Value Set Created with Add Segment
Independent
Independent
Independent
Independent
Format Only
Format Only
Format Only

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

## Usages

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

#### **Parameters**

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

#### **Delimiters**

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

#### Initial Values

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

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use the following bind variables in the WHERE clause of the SQL statement.
  - :{SEGMENT.segment\_code>}: Identifies a segment in the same context.



- : (CONTEXT. < context\_code>; SEGMENT. < segment\_code>): Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it cannot be a multiple-row context.
- :{VALUESET.</br>
   value\_set\_code>}: Identifies the closest prior segment in the same context that is assigned to the specified value set.
- :{FLEXFIELD.<internal code>}: Identifies a flexfield.

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

### Business Intelligence

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

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

## Enabling Descriptive Flexfield Segments for Business Intelligence: Points to Consider

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

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

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

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

## Flattening

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

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

- If you assigned a label to the segment, the flattened components include an additional single attribute representing segments with that label.
- If you didn't assign a label, the flattened components include a discrete attribute for each BI-enabled contextsensitive segment in each context.

Mapping to Logical Objects in Business Intelligence

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



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

Non-labeled context-sensitive segments aren't equalized across context values, so the flattened components include a separate attribute for each context-sensitive segment for each context value. It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a global segment, context segment, or context-sensitive segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence.

Note: Assigning a label to a context-sensitive segment serves to equalize the attribute across contexts, as well

as map the equalized attribute to business intelligence.

#### Managing Labels

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

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

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

#### Importing to Oracle Business Intelligence Repository

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

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

Note: When you import a flexfield into the Oracle Business Intelligence repository, you see both <name>\_ and <name>\_ attributes for each segment, along with some other optional attributes. The <name> attribute contains the value. The <name>\_ attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.



# Manage Extensible Flexfields

## Extensible Flexfields: Explained

Extensible flexfields are like descriptive flexfields, with some additional features.

- You can add as many context-sensitive segments to the flexfield as you need. You aren't restricted by the number of columns predefined and registered for the flexfield.
- You can configure a one-to-many relationship between the entity and its extended attribute rows.
  - A row of data can have multiple contexts associated with it.
  - A row of data can have multiple occurrences of the same context.
- You can configure attributes in groups to form a context so that the attributes in the context always appear together
  in the user interface.
- You can use existing hierarchical categories so that entities inherit the contexts that are configured for their parents. Contexts are reusable throughout categories.
- Application development has registered some extensible flexfields to support view and edit privileges. For such
  flexfields, you can specify view and edit privileges at the context level to control who sees the attributes and who can
  change the attributes' values.

When you configure a context for multiple rows per entity, the segments are displayed as a table.

Unlike descriptive flexfields, the extension columns corresponding to extensible flexfields segments are part of extension tables, separate from the base application table. Unlike descriptive flexfield contexts, the set of attributes in an extensible flexfield context remains constant and doesn't differ by context value.

An extensible flexfield describes an application entity, with the run time ability to expand the database that implementation consultants can use to define the data structure that appears in the application.

Extensible flexfields support one-to-many relationships between the entity and the extended attribute rows.

To get a list of predefined extensible flexfields, open the Setup and Maintenance work area, and use the Manage Extensible Flexfields task.

The following aspects are important in understanding extensible flexfields:

- Usages
- Categories
- Pages
- Security
- Protected Extensible Flexfield Data

#### Usages

As with descriptive flexfields, you can define multiple usages for an extensible flexfield, which enables several application tables to share the same flexfield.

For example, a flexfield for shipping options can be used by both a Supplier table and a Buyer table. In addition, you can associate a context with one, some, or all of the flexfield's usages. Thus, with the shipping information example, you can



associate a warehouse context with the Supplier usage, a delivery location context with the Buyer usage, and a ship-via context with all usages.

Usages include security information for applying no security to user access or enforcing view and edit privileges. Some product-specific extensible flexfields have specialized usage fields beyond those for security.

#### Categories

You can configure multiple extensible flexfield contexts and group the contexts into categories. All extensible flexfields have at least one category. For some extensible flexfields, you can configure a hierarchy of categories. A child category in the hierarchy can inherit contexts from its parent category.

You can define categories for extensible flexfields, and you can associate any combination of contexts with a given category.

For example, the Electronics and Computers category hierarchy might include a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on. The Home Entertainment product might have contexts that specify voltage, dimensions, inputs and outputs. Contexts are reusable within a given extensible flexfield. For example, the dimensions context could be assigned to any category that needs to include dimensional information.

#### Pages

Extensible flexfields let you combine contexts into groups known as pages, which serve to connect the contexts so they will always be presented together in the application user interface.

Each application page corresponds to one extensible flexfield category, with a separate region of the page for each associated context.

## Security

When you configure a flexfield, you set the privileges for a context at the usage level by selecting actions for the view and edit privileges of a context usage.

When an end user performs a search, the user interface displays only the attribute values of the contexts for which the user has view privileges. The user is able to perform a search using all attributes for all contexts, regardless of view privileges.

If end users access a context through a web service, an exception is thrown if they perform an action for which they don't have privileges.

All extensible flexfields have a base data security resource. Some data security resources for extensible flexfields are preconfigured with actions that you can use to specify access privileges. If no action is preconfigured, a security administrator can create actions and policies to support access control on the extensible flexfield attributes.

Some extensible flexfields have a translatable option; these flexfields also have a translation data security resource.

#### Protected Extensible Flexfield Data

Application developers may mark some data configurations in an extensible flexfield as protected, indicating that you can't edit them.

If an extensible flexfield is partially protected, then you can't edit the protected portions of the flexfield's configuration. For example:

- If an extensible flexfield context is protected, you can't edit its:
  - Context details
  - Context segments
  - Context usages



- If an extensible flexfield page is protected, you can't:
  - Edit the page details or delete the page
  - Edit the contexts associated with the page

#### Note:

- There is no restriction on page references to protected contexts. Custom pages you create may contain any context, whether protected or not.
- There is a restriction on category references to protected contexts. If a context is protected, you can't add it to or delete it from any category.

## Planning Extensible Flexfields: Points to Consider

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users. Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list, Manage Sandboxes, and Highlight Flexfields for adding and editing flexfield segments.

Planning an extensible flexfield can involve the following tasks:

- 1. Identify a hierarchical structure of categories.
- 2. Identify existing context values.
- 3. Identify custom attributes and plan the extensible flexfield segments, segment properties, and structure.
- 4. Plan validation rules.
- 5. Plan initial values.
- 6. Plan security.
- 7. Plan attribute mapping to Oracle Business Intelligence objects.

#### Category Hierarchy Structure

Existing category hierarchy structures provide the framework for planning what segments to add to an extensible flexfield as custom attributes of an entity.

Some Oracle Fusion applications provide user interfaces to create and manage an extensible flexfield's category hierarchy.

## Contexts and Existing Context Values

If related custom attributes can be grouped together, plan adding the attributes as a context of segments, and plan the order in which the attributes should appear.

Some extensible flexfields have preconfigured context values. Region headers displayed in a the user interface page or pages that contain the flexfield segments identify existing contexts. Using the Manage Extensible Flexfields task, find and open the flexfield for editing to view the list of configured context values.

See product-specific information for guidance in using preconfigured context values.

#### Plan the Segments and Segment Properties

List all the custom attributes that you want to add as extensible flexfield segments.

For each segment, define properties, including the indexed property.



#### Plan Validation Rules

Define each segment's validation rules and check if value sets exist for those rules or you must create new ones. If you must create a value set, you can create it either before you configure the flexfield or at the same time that you create or edit a segment.

When determining a segment's validation rules, consider the following questions:

- What is the data type character, date, date and time, or number?
- Does the segment require any validation beyond data type and maximum length?
- Should a character type value be restricted to digits, or are alphabetic characters allowed?
- Should alphabetic characters automatically be changed to uppercase?
- Should numeric values be zero-filled?
- How many digits can follow the radix separator of a numeric value? In base ten numerical systems the radix separator is decimal point.
- Does the value need to fall within a range?
- Should the value be selected from a list of valid values? If so, consider the following questions:
  - Can you use an existing application table from which to obtain the list of valid values, or do you need to create a custom list?
  - o If you are using an existing table, do you need to limit the list of values using a WHERE clause?
  - Does the list of valid values depend on the value in another flexfield segment?
  - o Is the list of valid values a subset of another flexfield segment's list of values?

#### Plan Initial Values

For every segment, list the constant value or SQL statement, if any, to use for the initial value of the custom attribute.

#### Plan Security

Determine what privileges to set for view and edit access to context attributes, such as providing all end users with view access but only managers with edit access.

If your security restrictions apply to several contexts, you can create generic actions. At a minimum, create the generic actions for the base data security resource. If the flexfield has a translatable option and you plan to use translatable contexts, then also create the generic actions for the translation data security resource. For example, if the Item flexfield supports the translatable option and has a data security resource ITEM\_EFF\_VL in addition to the base data security resource ITEM\_EFF\_B, then create actions for both data security resources, such as EDIT\_NONTRANS\_ATTRS for ITEM\_EFF\_B and EDIT\_TRANS\_ATTRS for ITEM\_EFF\_VL.

If your security restrictions are more fine-grained, such as needing to secure each context with a different privilege, then you can create more fine-grained actions.

#### Plan Which Segments Map to Oracle Business Intelligence Objects

If an extensible flexfield has been enabled for Oracle Business Intelligence, you can make the attributes available for use in Oracle Business Intelligence applications.

## Managing Extensible Flexfields: Points to Consider

Configuring extensible flexfields involves managing the available flexfields registered with your application database.



The following sequence describes how to configure extensible flexfields:

- 1. Configuring contexts by creating each context segment and the context-sensitive segments for each context segment, and providing the following for each segments:
  - a. Identifying information
  - b. Column assignment
  - c. Initial default value
  - d. Display properties
- 2. Configuring context usages and usage security by selecting actions to which users should have access:
  - View
  - Edit
  - None, if no special privileges should be enforced.
- 3. Configuring categories and category details.
- 4. Associating contexts with a category.
- 5. Creating logical pages for a category.

The following aspects are important in understanding extensible flexfield management:

- Contexts and pages
- Categories
- Initial values
- Adding segments to highlighted extensible flexfields
- Indexed segments
- Security
- Deployment

#### Contexts and Pages

Each context is displayed to end users as a region containing its context-sensitive segments. You can specify instruction help text to display instructions that explain how to use the region and its attributes to end users. Instruction help text is displayed at the top of the context region. A context can be defined as single row or multi row. Single row contexts are the same as descriptive flexfields contexts. A single row context has only one set of context-sensitive segments. A multi-row context enables you to associate multiple sets of values with the same object instance.

For example, for a BOOK table, you could create a multi row context named chapters that contains a segment for chapter and a segment for number of pages. Multiple chapters can then be associated with each book in the BOOK table.

For contexts that store multiple rows, you can uniquely identify each row by having the values in each row form a unique key.

If flexfield has a category hierarchy, then you can leverage the hierarchy to reuse contexts for similar entities, such as similar items in a product catalog.

Set the context to translatable so that free-form text entered by end users is stored in the language of the user's locale, and different translations of that text can be stored in other languages. Segments in the translated contexts should utilize formatonly value sets for storing free-form, user-entered text.

Set the context security to give an end user view or edit access to a context. The context's task flow and region appear in the user interface only for users with view access. With edit access, an end user can edit the context's attribute values. With no action specified for a usage, no special privileges are enforced through the context's configuration.



Define logical pages to group contexts together in the user interface. For a given category, you may create one or more logical pages. You may add one or more of the category's associated contexts to each of the category's logical pages.

You can specify:

- The sequence of the contexts within each page.
- The sequence in which the logical pages appear.
- Instruction help text to display instructions that explain how to use the page to end users. Instruction help text is displayed at the top of the logical page, preceding all of its context regions.

#### Categories

A category is a grouping of related data items that can be considered to belong together. You can associate any combination of contexts with a given category. Extensible flexfields with more than 30 categories must be deployed as a background process.

A category hierarchy logically organizes a set of categories. For example, the Electronics and Computers category hierarchy might include a Computer category and a Home Entertainment category, which in turn might include an Audio category and a TV category, and so on.

A category can be a child or sibling of an existing category. The hierarchy can be as simple or as complex as desired, with any combination of zero or more sibling categories and zero or more child categories. If no category is defined, the data items are grouped under a single predefined default category.

Each category has associated contexts that store relevant information about a data item in that category. For example, a Home Entertainment product has contexts that specify Voltage, Dimensions, Inputs and Outputs. Contexts are reusable within a given extensible flexfield. Then, the Dimensions context could be assigned to any category that needs to include dimensional information.

If a hierarchy includes child categories, each child category inherits the contexts from its parent category; for example, the Home Entertainment category inherits Voltage and Dimensions from the Electronics and Computers category.

Each extensible flexfield is associated with a particular category hierarchy. Consider category hierarchies to be defining framework for extensible flexfields and their contexts. A category hierarchy specifies which contexts are valid for each category.

An extensible flexfield can include multiple contexts which you define to support a given category. These contexts can be suitable for various purposes, but within a particular category, some contexts might be considered to be related to, or dependent on, each other. You can combine these contexts into groups known as logical pages, and determine the sequence in which the pages appear. This serves to connect the contexts so they will always be presented together and in a particular order in the application user interface.

For example, the Home Entertainment category might have an Electrical Specifications page that contains the Voltage, Inputs and Outputs contexts, and a Physical Specifications page that contains the Dimensions and Form Factor contexts.

#### Initial Values

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

You can use two types of SQL statements:

- SQL statement with no binding. For example, select MIN(SALARY) from EMPLOYEES.
- SQL statement with bind variables. You can use the following bind variables in the WHERE clause of the SQL statement.



- :{SEGMENT.segment\_code>}: Identifies a segment in the same context.
- :{context.<context\_code>; segment.<segment\_code>}: Identifies a segment in a different context. The context must be in the same category or in an ancestor category, and it cannot be a multiple-row context.
- :{VALUESET.</br>
   :{VALUESET.
   value\_set\_code>}: Identifies the closest prior segment in the same context that is assigned to the specified value set.
- : {FLEXFIELD. < internal\_code>}: Identifies a flexfield.

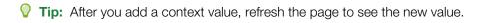
For more information about using bind variables, see the help for value sets.

#### Adding Segments to Highlighted Extensible Flexfields

When you highlight flexfields on a run time page and use an **Add Segment** icon button to create a segment, the segment code, name, description, table column, and sequence number are set automatically. If you use an **Add Segment** icon button to configure extensible flexfield segments, you can't use an existing value set. Value sets are created automatically when you add segments. You can enter the valid values, their descriptions, and the default value or specify the formatting constraints for the value set, such as minimum and maximum values.

Depending on display type, the value set you create with the **Add Segment** icon button is either an independent value set or a format-only value set. The table shows which type of value set is created depending on the segment display component you select.

Display Component	Value Set Created with Add Segment
Check Box	Independent
Drop-down List	Independent
List of Values	Independent
Radio Button Group	Independent
Text Field With Search	Independent
Text box	Format Only
Text area	Format Only
Rich Text Editor	Format Only
Date/Time	Format Only



## Indexed Segments

You can designate an extensible flexfield segment as indexed so that it's one of the selectively required attributes a user can use in an attribute search. If you indicate in the Manage Extensible Flexfield UI page that a segment should be indexed, the



column representing the segment must be added to the database index. Commonly, a database administrator (DBA) adds columns to the database index.

When an extensible flexfield with indexed segments is deployed, search task flows are generated along with the other flexfield artifacts and specify the indexed attributes as selectively required. In the deployed extensible flexfield's search task flow, an end user must specify at least one of the indexed attributes in the search criteria. This prevents non-selective searches, which could cause performance issues.

For example, if you index the memory and processor attributes and ensure that the corresponding columns in the database are indexed, a user can search an item catalog for computers by entering processor or memory or both as a search criteria. No search is performed if an end user enters an attribute that isn't indexed as a search criterion.

### Security

An extensible flexfield's base data security resource typically has a name with an \_B suffix. The translation data security resource is a view of a translation table that typically has a name with an VL suffix.

If a flexfield supports the translatable option and has a translation data security resource, make sure that you create the action for the appropriate data security resource.

- If you create a context-specific action for a nontranslatable context, add it to the base data security resource.
- If you create a context-specific action for a translatable context, add it to the translation data security resource.

### Deployment

You can only deploy extensible flexfields using the Manage Extensible Flexfields task. You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can queue up several extensible flexfields and deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command. When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a background deployment process. A flexfield's offline status is cleared and it's deployment status updated when the background deployment process has completed.

Note: The Offline Status column refreshes when you perform a new search in the Manage Extensible Flexfields task.

## Enabling Extensible Flexfield Segments for Business Intelligence: Points to Consider

An extensible 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's available for use in Oracle Business Intelligence.

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

- Flattening business components to use BI-enabled segments in Oracle BI
- Mapping attributes of flattened business components to logical objects in Oracle BI

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



#### Flattening

When you deploy a business intelligence-enabled extensible flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn't assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

## Importing to Oracle Business Intelligence Repository

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence and then propagate the flexfield object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION\_APPS\_BI\_APPID user. To import flexfield changes into the Oracle Business Intelligence repository in Oracle Cloud implementations, run the Import Oracle Fusion Data Extensions for Transactional Business Intelligence process. For additional information about import, refer to the Oracle Transactional Business Intelligence Administrator's Guide.



**Tip:** When you import a flexfield into the Oracle Business Intelligence repository, you see both <name>\_ and <name>\_ attributes for each segment, along with some other optional attributes. The <name>\_ attribute contains the value. The <name>\_ attribute contains the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

## Managing Extensible Flexfield Categories: Points to Consider

Categories are a way of extending the number of context-sensitive segments for a flexfield beyond the columns reserved for flexfield segments.

An Items extensible flexfield has a category for each item and each category can have one or more contexts. The laptop item belongs to the Computers category. Since extensible flexfields are mapped to separate extension tables, not just to columns as with descriptive flexfields, the thirty reserved columns on the extensible flexfield table let you define up to thirty context-sensitive segments for each context.

If you add a Dimensions context to the Computers category, thirty segments are available. But if you need to add more than thirty attributes, create another context and associate it to the same category. You could now add an Electronics Attributes context to the same Computers category in which you create another thirty segments.

You can continue creating more contexts and adding them to the Computers category. In this way your laptop computer item can be extended with as many attributes as you need, because it is mapped to a category and you can keep adding contexts to that category.

A descriptive flexfield on an items table with thirty columns reserved for segments can only have a single context. Once you configure the columns for that one context, you cannot create any more segments.

#### Predefined and Preconfigured Categories

How you structure the flexfield configuration depends on how categories are defined for the flexfield. If the extensible flexfield is preconfigured with one category, associate all your contexts and pages with that category. If a product-specific extensible flexfield is preconfigured with several categories, associate your contexts and pages with those categories. If the extensible flexfields provide user interfaces for configuring multiple categories, associate a context with more than one category using inheritance.

Some products provide an activity or task for creating and maintaining categories for an extensible flexfield. See product-specific information to determine if you can create categories for the flexfield.



You can view a flexfield's category hierarchies by using either the Highlight Flexfields feature or the Manage Extensible Flexfields task to find and open the flexfield for editing.

## Disabling Categories

While configuring an extensible flexfield, you can disable a category. The Enabled column in the Category table of the Edit Extensible Flexfield page, indicates which categories are enabled.

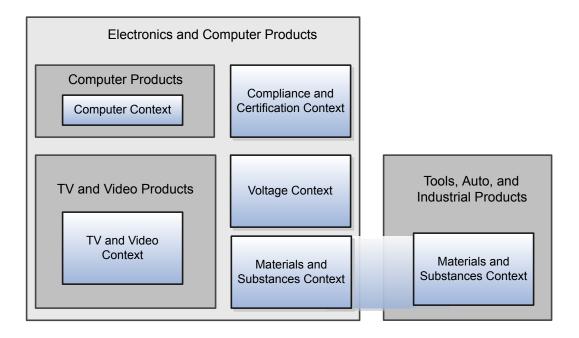
Note: When you deploy an extensible flexfield that has a disabled category, that category and its descendant categories aren't deployed. Contexts and their segments are deployed only if they belong to at least one enabled category.

#### Contexts

Group similar custom attributes into contexts. The group is displayed together in a region. The region's header is the context value.

If a category hierarchy exists for the flexfield, then you can leverage the hierarchy to reuse contexts for similar entities, such as similar items in a product catalog.

The figure shows the Item Extended Attributes flexfield, which uses the category hierarchy feature to reuse contexts. The flexfield's Electronics and Computers category contains contexts for compliance and certification, voltage, and materials and substances. The TV and Video subcategory and the Computer Products subcategory inherit the Electronics and Computer contexts in addition to having their own contexts. The Materials and Substances context belongs to both the Electronics and Computer Products category and the Tools, Auto, and Industrial Products category.



The table shows an example of category hierarchy for an extensible flexfield.

Display Name	Code	Description
Electronics and Computers	PROD_ ELECTRONICS	Electronics and Computers



Display Name	Code	Description
TV and Video	PROD_TV_VIDEO	Television and Video
Computers	PROD_COMPUTERS	Computers
Office Products and Supplies	PROD_ OFFICE_ PRODUCTS_ SUPPLIES	Office Products and Supplies
Tools, Auto, and Industrial	PROD_TOOLS_AUTO_INDUSTRIAL	Tools, Automotive, and Industrial
Sports and Outdoors	PROD_SPORTS_OUTDOORS	Sports and Outdoors

To store voltage information for all electronic and computer items, associate a Voltage context with the Electronics and Computers category. Both the TV and Video subcategory and the Computers subcategory then inherit the Voltage context from the parent Electronics and Computers category.

## Configuring an Item Extended Attributes Flexfield: Example

The Item Extended Attributes flexfield provides segments for extending the Item business object. In the Manage Extensible Flexfields task, you configure your product business object to include a Technical Specifications logical page in the user interface for the Electronics and Computers category of items.

In this example, your configuration of this flexfield groups custom attributes into the following contexts:

- Materials and Substances
- Compliance and Certification
- Voltage

### Scenario

The following list shows an example plan for custom computer attributes for the Item Extended Attributes flexfield. In this example, the Electronics Information page is inherited from the parent Electronics and Computers category.

- Page: Electronics Information
  - o Context: Compliance and Certification, single row
    - ISO 14001 (International Organization for Standardization for an Environmental Management System)
    - ENERGY STAR (energy efficiency guidelines)
    - ROHS (Restriction of the use of certain hazardous substances in electrical and electronic equipment)
  - Context: Voltage, single row
    - Minimum voltage
    - Maximum voltage
    - Current type
  - Context: Materials and Substances, multiple rows
    - Material
    - Contain recyclate
    - Percent unit mass



- Page: Computer Information
  - Context: Processor Specifications, single row
    - Manufacturer
    - CPU type
    - Processor interface
    - Processor class
    - Processor speed
    - Cores

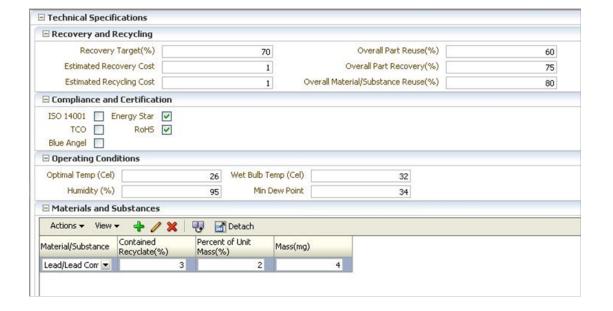
The following table summarizes key decisions for this scenario.



## Collecting Technical Specifications

Your product inventory pages for electronics and computers require a technical specifications page. Your product inventory pages for furniture require a furniture specifications page and an assembly instructions page. Items in both the electronics and computer category, and in the furniture category, share attributes for specifying materials and substances.

The figure shows a Technical Specifications logical page in the user interface for the Electronics and Computers category, with attributes in the context of Recovery and Recycling, Compliance and Certification, Operating Conditions, and Materials and Substances. The Materials and Substances context is configured for multiple rows so your users can select all the materials and substances required to make a single product, displayed as attribute values in a table.





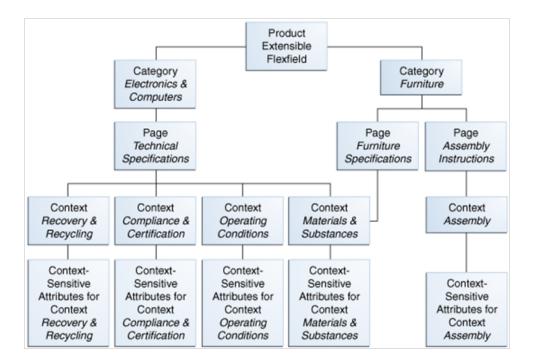
## **Analysis**

You use logical pages to arrange how the contexts appear in the user interface. Use a context to store all the materials and substances required to make a single product. You can configure a context to store multiple rows per entity. The multiple rows are displayed in a table, as for the Materials and Substances context.

The Technical Specifications logical page contains the attributes for the four contexts.

- Recovery and Recycling
- Compliance and Certification
- Operating Conditions
- Materials and Substances

In the figure, the Furniture category is configured to include a Furniture Specifications logical page and an Assembly Instructions logical page. The two categories (Electronics & Computers and Furniture) share the Materials & Substances context.



## Configure Security for the Item Flexfield Configuration

The following table shows an example of data security policies for the Item flexfield.

Data Security Resource	Policy	Role	Actions	Condition
ITEM_EFF_B	А	VOLTAGE_SPEC	edit_nontrans_ voltage_ctx	All values
ITEM_EFF_VL	В	COMPLIANCE_ SPEC	edit_ trans_ compliance_ ctx	All values



Data Security Resource	Policy	Role	Actions	Condition
ITEM_EFF_VL	С	COMPUTER_SPEC	edit_ trans_attrs	ComputerCategoryFilter
ITEM_EFF_VL	D	TELEVISION_ SPEC	edit_ trans_attrs	TVCategoryFilter

The following table shows the privileges for three of the flexfield's contexts.

Context	Edit Privilege	View Privilege
Voltage	edit_ nontrans_ voltage_ctx	NONE
Compliance and Certification	edit_ trans_ compliance_ ctx	NONE
Materials and Substances	edit_ trans_attrs	NONE

In this example, anyone can view the contexts' attributes, but the edit privileges are restricted as follows:

- Voltage: Editable only by voltage specialists.
- Compliance and Certification: Editable only by compliance specialists.
- Materials and Substances: Only computer specialists can edit these attributes for items in the computer category.
   Only television specialists can edit these attributes for items in the TV category.

In this example, the Materials and Substances context is secured by a generic action with a condition applied to restrict access by category. Voltage and Compliance and Certification are secured by actions specific to each context.

# FAQs for Manage Extensible Flexfields

## Why did the extensible flexfield context not appear at run time?

If a deployed extensible flexfield context doesn't appear in the user interface, verify that the context is associated with one of the category's pages defined for the extensible flexfield.

# Manage Key Flexfields

## Key Flexfields: Explained

Key flexfields provide a means to capture a key such as a part number, a job code, or an account code. A key flexfield consists of one or more segments, where each segment can have a meaning.

For example, a part number 10-PEN-BLA-450 might correspond to a black pen from supplier #450 sold by division #10 (office supplies). Behind the scenes, the application uses a unique number, 13452, for this part, but the user always sees the 10-PEN-BLA-450 part number.



The following aspects are important to understanding key flexfields:

- Architecture
- · Segments and segment labels
- Structures
- Segment and structure instances
- Combinations
- Dynamic combination creation
- Security

Key flexfields aren't optional. You must configure key flexfields to ensure that your applications operate correctly. You configure and maintain key flexfield definitions with the Manage Key Flexfields task. To get a list of predefined key flexfields, open the Setup and Maintenance work area, and use the Manage Key Flexfields task. For information about specific key flexfields, see the help for the product where the associated business component is implemented.

#### Architecture

Flexfield metadata is stored in the flexfield metadata tables. When you configure a key flexfield, you define metadata about the key flexfield covering aspects such as:

- Segments are in a structure
- · Structures in the flexfield
- · Value sets in each segment

Based on the flexfield metadata, actual part numbers are captured at run time as a combination of segment values and stored in a combinations table. A combinations table contains all the segment columns for a flexfield, a unique ID column, and a structure instance number column. The structure instance number column differentiates multiple arrangements of the segment columns. For example, a part number containing multiple segments can be represented by a key flexfield. A part number key flexfield has a corresponding combinations table. In that table, the flexfield stores a list of the complete codes, with each segment of the code in a column, with the corresponding unique ID and structure instance number for the code. When users define a new part number or maintain existing part numbers in the parts catalog, they directly maintain rows in the combinations table.

The foreign key table contains a different business entity than the combinations table. For example, the business entity in the foreign key table is order lines or invoice lines that contain foreign key references to parts for ordering. Any number of foreign key tables can reference a particular entity represented by a key flexfield.

#### Segments and Segment Labels

A key flexfield contains segments and a segment label identifies a particular segment within a key flexfield. Segment labels are defined and made available by the product development. A segment contains the following details:

- A prompt
- A short prompt
- Display width
- The sequential position of the segment within the key flexfield structure
- The range type
- · Column name of the attribute being stored by the segment
- A default value set
- A label for the segment



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, the requirement is to identify which segment in the accounting flexfield contains balancing information and which segment contains natural account information. A segment label determines 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 helps a user searching for segments, such as the Cost Center label for all segments across key flexfields that store a value for the cost center.

#### Structures

A key flexfield structure definition includes the number of segments and their order.

In some applications, different users like 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 users based on a data condition in your application data, such as the value of another field entered by the 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 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. To store a particular segment, such as Cost Center, in two different structures, you must define the segment separately in each structure. 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 values or value sets used in validating 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 is registered with a tree structure, you can specify a tree code for a segment instance.

#### Combinations

A combination is a complete code, or combination of segment values that makes up the code, that uniquely identifies an object.

For example, each part number is a single combination, such as PAD-YEL-11x14 or 01-COM-876-7BG-LTN. In these combinations, the hyphen is the segment separator. If you have ten parts, define ten combinations. A valid combination is an existing or new combination that can be used because it's currently active and doesn't violate cross-validation or security rules. A combination has different segments depending on the flexfield structure being used for that combination. Any combination is associated with only one particular flexfield structure.

Many applications refer to a key flexfield combination by using the name of the entity or the key flexfield itself. For example, Assets uses the asset key flexfield and refers to one of its combinations as an asset key or asset key flexfield. In another example, Oracle Fusion General Ledger refers 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 (an account combination ID) for that code. Then, other tables in the application have a column that stores just the unique ID for the code. For example, you may have a part number code, such as PAD-YEL-11x14. The Parts combinations table stores



that code along with its ID, 57494. If your application lets you take orders for parts, you might then have an Orders table that stores orders for parts. That Orders table would contain a single column that contains the part ID, 57494, instead of several columns for the complete code PAD-YEL-11x14. Typically, one combinations page maintains the key flexfield, where the key flexfield is the representation of an entity in your application. Maintain individual combinations, such as part numbers in the combinations page.

## Dynamic Combination Creation

Dynamic combination creation is the insertion of a new valid combination into a combinations table from a page other than the combinations page. Dynamic combination creation may be enabled at the following levels.

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 don't permit dynamic combination creation, you may control whether dynamic combination creation is enabled for each structure instance. If enabled, a user can enter a new combination of segment values using the flexfield window from a foreign key page. For example, when entering a transaction, a GL user can enter a new expense account combination for an account that doesn't yet exist. Your application creates the new account by inserting the new combination into the combinations table behind the scenes. Assuming that the new combination satisfies any existing cross-validation rules, the flexfield inserts the new combination into the combinations table, even though the combinations table isn't the underlying table for the foreign key page.

## Planning Key Flexfields: Points to Consider

Your first step in planning your key flexfields is to determine which key flexfields your application requires.

Your plan should include:

- The purpose of the key flexfield
- The number and length of its available segment columns
- Whether your key flexfield allows more than one structure
- Whether more than one structure must be defined
- The number, order and length of your segments for each structure

Consider the following aspects in planning flexfields:

- Before you begin
- · Access to flexfield-related tasks
- Restrictions
- Validation rules for flexfield segments



## Before You Begin

Once you have identified a flexfield to configure, plan the configuration in advance. Compile a list of the UI pages and other artifacts in your deployment that are affected by the configuration. Verify that you are provisioned with the roles needed to view and configure the flexfield. View the flexfield using the Highlight Flexfields command in the Administration menu while viewing the run time page where the flexfield appears. Plan how you will deploy the flexfield for test and production users.

Review the tools and tasks available for managing flexfields, such as the Define Flexfields task list and Manage Sandboxes.

If you plan to use value sets, create them before configuring the key flexfield. You cannot create value sets for key flexfields at the time that you add and configure key flexfield segments.

#### Access to Flexfield-Related Tasks

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

#### Restrictions

If you plan to use value sets, create them before configuring the flexfield.

Plan your key flexfield configuration to scale to your enterprise needs. For example, if you expect to disable old cost centers and enable new ones frequently, plan a larger maximum size for your cost center value set so that you can have more available values. One thousand available values for a 3-character value set provides more room for changes than 100 available values for a 2-character value set.

Note the code name of the flexfield you intend to configure so you can find it easily in the Define Flexfield task list or tasks for managing product-specific key flexfields.

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

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

### Reporting

If you want to report on your data by certain criteria or sub-entities, such as account number or project or region, consider making that sub-entity a distinct segment, rather than combining it with another sub-entity, so that you can categorize and report on smaller discrete units of information.

## Managing Key Flexfields: Points to Consider

Consider the plans for a key flexfield, security, and resulting run time pages when configuring key flexfields.

### **Planning**

Plan structures carefully and allow for future needs. Don't change the number, order, and maximum length of segments once you have acquired flexfield data.

#### Structure Delimiters

A delimiter separates the segments when they appear to end users. The delimiter value of a structure specifies the character used to visually separate segment values when the key flexfield is displayed as a string of concatenated segments in the UI.

Choose the delimiter value of your key flexfield carefully so that it doesn't conflict with the flexfield data. For example, if your data frequently contains periods, such as in monetary or numeric values, don't use a period as your segment separator. Any character you expect to appear frequently in your segment values or descriptions isn't a good choice for the delimiter. If you



change the configuration of a key flexfield, such as the delimiter, the change affects the previously stored key flexfields with that structure.

#### Security

Oracle Fusion data security enforces value set security.

Within key flexfields, value set security applies to the selection of the individual segment values in the segment list of values. When selecting a key flexfield segment value from the combinations table, data security allows display of only the combinations whose segment values you have access to. Applications development controls whether or not value set security rules propagate to the foreign key table. By default they do.

### Run Time Pages

Application development determines the user interface (UI) pages used to render flexfields. The types of key flexfield UI pages are as follows:

- Combinations pages where the underlying entity objects use the combinations table itself
- Foreign key pages where the underlying entity objects contain a foreign key reference to the combinations table
- Partial usage pages where some or all of the key flexfield's segment columns are in a product table

The same key flexfield can be used in different ways on different pages.

A page with a foreign key reference has a base table or view that contains a foreign key reference to a combinations table with the actual flexfield segment columns. This lets you manipulate rows containing code combination IDs (CCID).

A page with partial usage of a key flexfield presents segments that are defined on a product's transactional table in addition to being defined on a combinations table. In the case of a partial usage page, it is possible that only part of the configuration is visible. This enables the key flexfield to behave more like a descriptive flexfield.

A code combination maintenance page or combinations page presents the combinations table. This enables directly creating and maintaining code combinations. The combinations table contains all key flexfield segment columns and a unique ID column.

A typical application has only one combinations page. An application might not have a combinations page if it doesn't support maintenance by administrators.

A page containing a search region enables end users to select which attributes of the key flexfield view object to use as criteria to search for flexfield metadata.

For example, you can configure seven segments for the Account key flexfield. In a foreign key reference page, end users see the typical key flexfield picker with all seven segments where they can search for combinations. In a partial usage page using the same key flexfield, end users potentially could see only a single segment such as the Cost Center labeled segment, or they might see multiple segments but displayed as individual segments rather than as a picker for choosing combinations.

For more information on key flexfield pages, see the Oracle Fusion Applications Developer's Guide.

# Key Flexfield Structures: Explained

A key flexfield structure arranges the segments of a key so that you can reuse a single key flexfield in multiple combinations of the same segments or a subset of those segments. Multiple instances of a single structure can accommodate differences in the value sets assigned to the structure's segments.



The structure determines the following aspects of a key flexfield:

- The segments to include
- · The order of the segments
- Segment labels on the included segments
- · Properties for each segment applied to the instances of the segments in an instance of the structure

### Managing Key Flexfield Structures

All the segments defined for a key flexfield are available to be included in a key flexfield structure.

You can define as many segments as there are defined segment columns in your key flexfield combinations table. Ensure that you add segments in the order that your key requires. Once deployed, the order cannot be changed.

Enable segments to indicate that they are in use. A flexfield doesn't display disabled segments in run time. 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

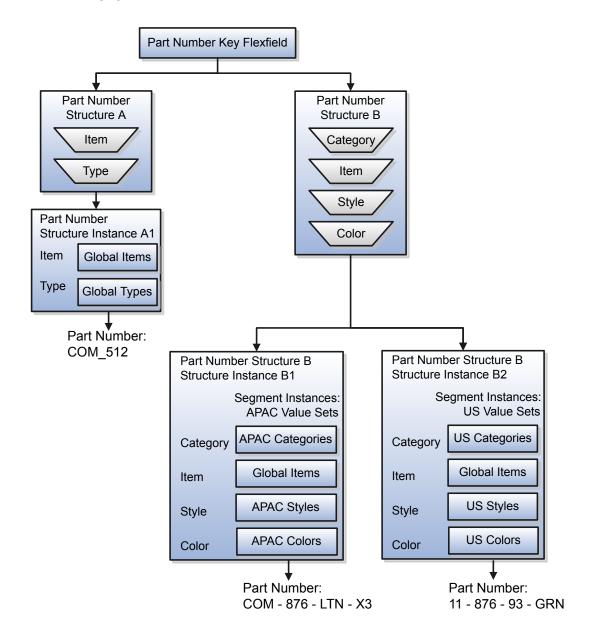
The differences among structure instances include whether dynamic combination creation is allowed. Likewise, at the structure instance level, differences among segment instances are based on the following:

- Value set
- Default type and default value
- Tree code
- · Whether the segment is any of the following:
  - Required
  - Displayed
  - Enabled for business intelligence
  - o Optional or required as a query criterion

For example, you can use one group of value sets for the US and another for France.



The following 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 defined for that structure. However, the structure instances may vary if the properties are defined at the structure instance or segment instance level. For example, the value set assigned to the segment instances.

## Query Required Segment Instances

You can designate a key flexfield segment instance as query required to make it a selectively required attribute. A user can use it a key flexfield combination search. If you indicate on the Manage Key Flexfields UI page that a segment instance requires indexing, add the column representing the segment to the database index. Commonly, a database administrator (DBA) adds columns to the database index.



Following deployment, the combination picker of the key flexfield displays the guery required attributes as selectively required. An user must specify at least one of the query required attributes in the search criteria. This prevents unnecessary searches that could cause performance issues.

For example, you mark the cost center and account attributes as query required and ensure that the corresponding columns in the database are indexed. A user can search for combinations by entering cost center or account or both as search criteria. No search is performed if a user doesn't enter at least one query required attribute as search criteria.

Vip: Index the Structure Instance Number column on your combinations table to improve run time performance.

#### Dynamic Combinations

If a key flexfield supports dynamic combination creation, you can select to enable this feature by selecting **Dynamic** Combination Creation Allowed. As a result, users enter values at run time that produce new account combinations for the flexfield. If Dynamic Combination Creation Allowed isn't enabled, new valid combinations can only be entered using the combinations table for the flexfield.

#### **Trees**

You may define a tree code for the value set assigned to the segment instance. When 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. The tree structure may be fixed across all segments in the flexfield, or may vary across segments.
- A tree code for that tree structure exists.
- The tree code includes tree versions containing the values of the value set assigned to the segment instance.
- You assign the required tree code directly to the segment instance.

If these conditions are satisfied, different segment instances that use the same value set can be assigned the same or different tree codes. They use a different hierarchy definition over the same values.

# Cross-Validation Rules: Explained

You can control the creation of new key flexfield code combinations by defining cross-validation rules. A cross-validation rule defines validation across segments and enforces whether a value of a particular segment can be combined with specific values of other segments to form a new combination.

The table compares segment validation to cross-segment validation:

Type of validation	Type of control
Segment validation	Controls the values you can enter for a particular segment
Cross-segment validation	Controls the combinations of values that administrators and end users can create for key flexfields



Note: You can use cross-validation rules for any key flexfield that has cross-validation enabled. See the documentation for your key flexfield to determine if it supports cross validation.



Cross-validation rules prevent the creation of combinations with values that shouldn't coexist in the same combination. For example, your company requires that all revenue accounts must have a specific department. Therefore, account combinations that have revenue account values, such as all values between 4000 and 5999, must have a corresponding department value other than 000, which indicates no department is specified. You can define cross-validation rules that disallow creation of combinations with incompatible segments, such as 4100-000 or 5000-000.

Alternatively, suppose your accounting key flexfield has an Organization segment with two possible values, 01 and 02. You also have a Natural Account segment with many possible values, but company policy requires that Organization 01 uses the natural account values 001 to 499 and Organization 02 uses the natural account values 500 to 999. You can create cross-validation rules to ensure that users cannot create a general ledger account with combinations of values such as 02-342 or 01-750.

The following aspects are important to understanding cross-validation rules:

- Rule Definitions
- Enforcement
- Timing

#### **Rule Definitions**

Cross-validation rules consist of the following information:

Rule Feature	Purpose
Name	Uniquely identifies cross-validation rules in a deployment.
Description	Helps administrators identify the purpose of the rule.
Error message	Explains why the attempted combination violates the rule.
Start Date, End Date	Indicates the period of time when the rule is in effect.
Enabled	Determines whether the rule is enforced.
Condition filter	Determines the conditions under which an enabled cross-validation rule should be evaluated.
Validation filter	Determines the validation that the rule enforces when that condition is met.

When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created. If the event specified in the condition filter isn't applicable, then the combination is considered to pass the rule and the rule won't be evaluated even if it is enabled.



#### Enforcement

Cross-validation prevents creation of invalid combinations by administrators using maintenance pages and end users using dynamic insertion in foreign key pages.



Enabled rules are enforced when there is an attempt to create a new combination of segment values. Disabled rules are ignored. Deleting the rule has the same effect, but you can re-enable a disabled rule.

#### **Timing**

When users attempt to create a new combination, the key flexfield evaluates any cross-validation rules that are enabled and in effect.

Note: Cross-validation rules have no effect on combinations that already exist. The flexfield treats any existing invalid combinations that pre-date the rule as valid.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, manually disable those combinations using the combinations page for that key flexfield.

When defining a cross-validation rule, specify a start and end date to limit the time when the rule is in effect. The rule is valid for the time including the From and To dates.

#### Cross-Validation Rules: Points to Consider

When you need key flexfield combinations of segment values validated across segments, you can optimize your cross-validation rules to improve the experience of administrators and end users.

Consider the following when defining cross-validation rules:

- Filters
- Rule Complexity
- Maintenance

#### **Filters**

A cross-validation rule includes a condition filter and a validation filter.

The rule is evaluated using the following logic: If the condition filter is satisfied, then validate that the validation filter is satisfied.

- 1. The condition filter describes the event under which the rule will be evaluated. If the event specified in the condition filter isn't applicable, then the rule won't be evaluated even if it is enabled.
- 2. When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created.

For example, if your organization has determined that a certain company value, Operations, cannot use a specific cost center, Marketing, you can define a cross-validation rule to validate your combinations.

- 1. The rule evaluates the company condition filter.
- 2. When company is equal to Operations, the rule evaluates the cost center validation filter.
- 3. When cost center is equal to Marketing, the rule prevents a combination from being created.
- **4.** The error message you defined for the rule displays to inform the user that the attempted combination violates the rule.
- Note: This rule doesn't affect the creation of combinations with Marketing cost center and company values other than Operations.

#### Rule Complexity

For optimal performance and ease of understanding, define several simple validation rules instead of using one complex rule. Simple validation rules let you provide a more specific error message and are easier to maintain over time.



Avoid rules that control validation across more than two segments, where possible. While you can define cross-validation rules that span two or more segments, keep in mind that it becomes more difficult to interpret cross-validation error messages and correct invalid key flexfield combinations as your rules encompass more segments.

#### Maintenance

To maintain consistent validation, review existing key flexfields when you update your cross-validation rules. Regardless of your current validation rules, Oracle Fusion Applications accept a key flexfield combination if the combination already exists and is enabled. Therefore, to ensure accurate validation, you must review your existing combinations and disable any combinations that don't match the criteria of your new rules.

▼ Tip: To keep this type of key flexfield maintenance to a minimum, decide upon your cross-validation rules when you first set up your key flexfield structure. Define cross-validation rules before creating combinations and before combinations are used in transactions.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, disable those combinations using the combinations page.

## Creating a Cross-Validation Rule: Example

Create cross-validation rules to prevent specific combinations of segment values in account combinations. For example, you can prevent a particular cost center from being combined with a specific company value. Cross-validation rules only affect the creation of new account combinations.

#### Scenario

Your company, InFusion America Inc. doesn't have a marketing department. Create a cross-validation rule to prevent InFusion America Inc. company value 01 from being combined with marketing department value 300 in an account combination.

- 1. Navigate to the Setup and Maintenance work area. Search for and select the Manage Cross-Validation Rules task.
- 2. Select the chart of accounts for InFusion America and click **Add Row**.
- 3. Enter a unique rule name, such as **IFAM01** and a description, such as **Do not combine Marketing department**, 300 with InFusion America company 01.
- 4. Specify an effective starting date of today and enable the rule.
- 5. Click the **Condition Filter** icon and enter **Company** equal to **01**. The cross-validation rule evaluates if company 01 is entered, and if so, then the validation process continues to evaluate the rule.
  - Note: If you don't specify a statement in the condition filter, then the rule is always evaluated.
- 6. Click the **Validation Filter** icon and enter **Cost Center** is not equal to **300**. When the rule is evaluated, an account combination must contain a cost center other than 300 before it can be created.
- 7. Enter the error message: Cost Center 300 is not allowed with Company 01. The message displays in the relevant user interfaces and processes when an account combination can't be created because it violates the rule.
- 8. Click Save and Close.

# Enabling Key Flexfield Segments for Business Intelligence: Points to Consider

A key flexfield 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's available for use in Oracle Business Intelligence.

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

- Flattening business components to use BI-enabled segments in Oracle BI
- Equalizing segments to prevent duplication and complexity in the flattened component



- Mapping attributes of flattened business components to logical objects in Oracle BI
- Managing the labels that map segments to logical objects in Oracle BI

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

#### Flattening

When you deploy a business intelligence-enabled key flexfield, the deployment process generates an additional set of flattened business components for use in business intelligence. The flattened business components include attributes for business intelligence-enabled segment instances only.

If you assigned a label to a segment, the flattened components include a single attribute representing all segment instances with that label. If you didn't assign a label, the flattened components include a discrete attribute for each BI-enabled segment instance in each structure.

#### Mapping to Logical Objects in Business Intelligence

You can simplify reporting by representing similar segments as a single logical object in Business Intelligence. If you assign a label to segments that serve the same purpose in different structures, you can consolidate the segments into a single attribute. This prevents duplication and the extra workload and complexity that result from the flattening process. For example, an organization may have more than one definition of its key accounting flexfield to support different requirements for accounting reporting. A US Accounting Flexfield structure may have a segment called Subaccount to track project expenditures. The same type of information may be tracked in a UK accounting flexfield structure with a segment called Project. Equalize these two segments to create a single list of values for reporting.

Non-labeled segments aren't equalized across context values, so the flattened components include a separate attribute for each segment for each structure. It may not be possible to equalize similarly labeled segments if they have incompatible data types or value set types.

Assign a label to a segment to map the corresponding attribute in the flattened components to a logical object in Oracle Business Intelligence. Using labels to map segments to BI logical objects minimizes the steps for importing the flexfield into Oracle Business Intelligence. 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 labels for assignment, as needed. Specify a code, name, and description to identify each label. In the BI Object Name field, enter the name of the logical object in Oracle Business Intelligence to which the segment label should map during import. Specifying the BI logical object minimizes the steps for importing the flexfield into Oracle Business Intelligence and helps to equalize context-sensitive segments across structures.

If no labels are assigned to a BI-enabled segment, or the BI Object Name on the assigned label doesn't exist in business intelligence, you must manually map the segment to the required 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.

#### Importing to Oracle Business Intelligence Repository

After you deploy a business intelligence-enabled flexfield, import the flexfield changes into the Oracle Business Intelligence repository to make use of the newly flattened business components in business intelligence. Then propagate the flexfield



object changes. When you import the metadata into the Oracle Business Intelligence repository, you must do so as the FUSION APPS BI APPID user.

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

Note: When you import a flexfield into the Oracle Business Intelligence repository, you see both <name>\_ and <name>\_ attributes for each segment, along with some other optional attributes. The <name>\_ attribute contains the value. The <name>\_ attribute contains the code of the value set that the value comes from, and is used for linking to the value dimension. You must import both attributes.

## Key Flexfields: Example

A key flexfield can capture expense account information.

#### Scenario

When entering details for each expense, the user specifies an account to which the expense is charged.

## **Entering Expense Accounts**

A user interface for entering expenses gives the user the option of selecting an expense account that identifies the cost center and other details needed for processing the expense.

## **Analysis**

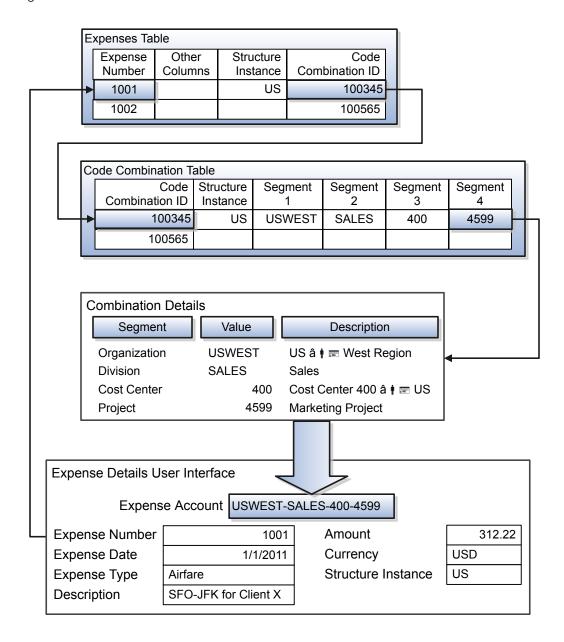
The expense account field is a foreign key reference to a code combination (EXPENSE\_LINES.EXPENSE\_ACCOUNT = ACCOUNTS.CCID).

# Code Combinations Table for Entering Accounts and Employees

The code combinations table supports entering account information, such as for expense accounts.



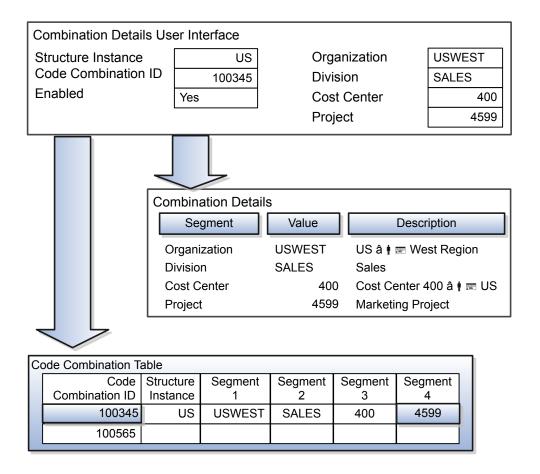
The figure shows the origin in the code combinations table of the account specified by the user. The code combination ID record stores the information of the key flexfield segments used to assemble the expense account based on the key flexfield configuration.



The combinations page, which is the maintenance page for the key flexfield, is for managing rows in the combinations table. In this example, managing the combinations means adding or editing account numbers that adhere to the key flexfield metadata rules.

The figure shows the code combination details for the example expense account reflected in the flexfield configuration and the code combinations table.





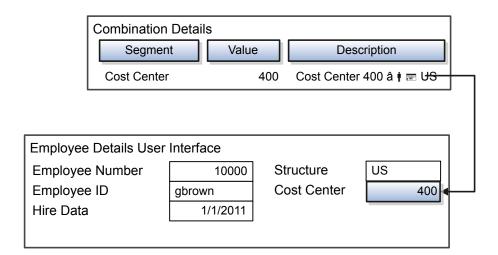
If dynamic combination creation isn't enabled, then when entering an expense line, the user can only select an account that already exists in the ACCOUNTS (combinations) table. If they require an account that doesn't exist, they must consult with the appropriate application administrator who can add the account to the combinations table.

If dynamic combination creation is enabled, then when entering an expense line, the user can either select a pre-existing account, or type in a new account that is created dynamically on the fly in the ACCOUNTS (combinations) table. Once the new combination is created, the same user can refer to it on the expense line.

When managing employee information, the user specifies the cost center that the employee belongs to. The cost center field corresponds to a single, labeled segment of the Account Key Flexfield and has metadata defined such as the allowable value set for that segment.

In this figure, instead of specifying a cost center ID reference to an account, only the Cost Center segment is used and the value is stored directly on the employee table.





#### Related Topics

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

# **Define Attachments**

# Attachments: Explained

You can use attachments to provide supplementary information to specific business objects. Attachments can be URLs, desktop files, text, or repository folders. For a business object you may view, create, delete, or edit attachments, depending on your role and granted privileges. For more information on 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 can't interact with the repository unless the repository mode is enabled. When enabled, users can share attachments among objects, update attachments, and perform other tasks. Access to the attachment files is controlled by a digital signing mechanism.

# Security

Data security applicable to a specific business object extends to its attachments For example, if a user has no access to a specific expense report, then that user cannot access its attachments. You can also use attachment categories to control access and actions on attachments, based on roles associated with that 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 with which attachments can be associated. Each attachment UI must be defined with a corresponding attachment entity. Attachment entities are used only in the context of attachments and exist separately from the database entities that they are based on.

In the Setup and Maintenance work area, search for the Manage Attachment Entities task. Use the Manage Attachment Entities page to edit and create attachment entities. You can either use the predefined attachment entities with attachment Uls or create entities, for example when developing custom Uls.

The entity name should match the name of the table or view that represents the business object used for attachment. The name is also used in the repository folder that is automatically created to store attachments for the entity.

The data security policies associated with the database resource defined for the attachment entity apply to attachments for that entity. However, the security setting must be enabled for that entity. 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.

#### Related Topics

- Modules in Application Taxonomy: Explained
- Database Resources and Data Security Policies: How They Work Together

# Attachment Entities and Attachment Categories: How They Work Together

The association between attachment entities and categories determines the use of categories for an entity. For example, categories associated with the expense report attachment entity are available in the attachment UIs for expense reports. You can configure the associations when managing either entities or categories. Between the Manage Attachment Entities and Manage Attachment Categories pages, any change in association on one page automatically reflects on the other page. You can open either page by starting in the Setup and Maintenance work area and searching for the attachment tasks.

# Managing Entities

On the Manage Attachment Entities page, you determine which attachment categories are relevant to a particular entity. Each entity must have at least one category. For a particular expense report page with attachments functionality, you can specify which category to use for the attachment. Accordingly, the data security defined for each category is applied to the attachments on that page if security is enabled.

# Managing Categories

If you create an attachment category and must assign it to multiple attachment entities, use the Manage Attachment Categories page. The association is the same as that on the Manage Attachment Entities page.

# Attachments Troubleshooting: Explained

Attachments UIs are very user-friendly and easy to work with. You may encounter issues in certain cases such as you customize the attachments, for example create additional attachment categories, or implement data security on them.



## Issue: Can't View, Add, Update, or Delete Attachments

You may encounter the following issues when trying to view attachments or perform actions such as adding attachments.

- You can no longer see specific attachments that were earlier visible.
- You can no longer update or delete attachments.
- You get an error stating that you 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. 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, 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 for the expense reports.

For more information on attachment category data security, see the Oracle Fusion Applications Developer's Guide.

Certain attachments UI have predefined restrictions for users on categories. 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

You can view existing attachments but the attachments no longer have an attachment category associated with them.

#### Resolution

When the attachment was added, at least one category existed for the corresponding attachment entity. 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 may need to check with your system administrator or help desk to determine the exact entity used on the page with the expenses attachments or any additional categories to assign.



Certain attachments UI have predefined restrictions for users on categories. 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?

You must use an attachment category to classify and secure an attachment. While adding attachments, you can view the available attachment categories and add the attachment to one of them. 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 restrict user access and actions for an attachment entity. You can also create and manage custom categories for your own purpose, involving specific attachments with specific security requirements. For more information on attachment category data security, see the Oracle Fusion Applications Developer's Guide.

In the Setup and Maintenance work area, search for the Manage Attachment Categories task and access the Manage Attachment Categories page.

#### Related Topics

• Modules in Application Taxonomy: Explained





# 10 External Integration

# Web Services

## Web Services: Overview

Use web services to integrate web-based applications into your Oracle Applications Cloud. Web services expose business objects and processes to other applications through the use of open standards-based technologies.

The web services support development environments and clients that comply with the following open standards:

- Extensible Markup Language (XML)
- Simple Object Access Protocol (SOAP)
- Business Process Execution Language (BPEL)
- Web Services Description Language (WSDL)
- XML schema definitions (XSD)

Oracle Applications Cloud includes two types of web services:

- Application Development Framework (ADF) services
- Composite services

The following table describes the two types.

Web Service Type	Description
ADF services	ADF services usually represent business objects, such as employees or purchase orders. ADF services typically expose standard operations, such as create, update, and delete. However, for locally-persisted objects, ADF services are not limited to these operations.
	Examples of ADF services include:
	<ul> <li>Worker.changeHireDate - a service that updates the hire date of the worker business object.</li> </ul>
	<ul> <li>ProjectTask.createTask - a service that adds a task to the project task business object.</li> </ul>
Composite services	Composite services usually represent end-to-end business process flows that act on business events produced by the ADF services. Composite services orchestrate multiple object-based services, rules services, and human workflows. Examples of composite services include:
	<ul> <li>ProjectStatusChangeApproval.process - a service that accepts the change in project status.</li> </ul>
	<ul> <li>ScheduleOrchestrationOrderFulfillmentLineService.scheduleOrders - a service that schedules resources used to fulfill an order.</li> </ul>

For more information about web services, see the SOAP Web Services guide for your cloud services.



# Developer Connect

## Developer Connect: Overview

The Developer Connect portal provides information about the web services deployed to your Oracle Applications Cloud instance. You can use this information to integrate with or extend Oracle Applications Cloud and develop customized business solutions.

To open the Developer Connect portal, from the **Navigator** menu, select **Tools** - **Developer Connect**. This portal displays dynamic information of the web services, and the customization done to web services to integrate with Oracle Applications Cloud. You can synchronize the Developer Connect portal with your cloud instance to retrieve the latest web service information such as service attributes, operations, business objects, security policies, and WSDL files.

Use the Developer Connect portal to:

- View the summary of the web service information such as the business object that the service defines, life cycle status, and security policy.
- Discover the operations available for the selected web service, and the request and response payloads for each operation.
- View the hierarchy of the service data objects and know information such as the data type and whether it's a required field about custom attributes.
- Review the sample payload XMLs for the operations of the web service. You can add or edit sample payloads, and also delete custom sample payloads.

## Web Service Life Cycle: Explained

A web service goes through three phases in a life cycle:

- Active: A service is active when it's delivered the first time, until it's deprecated.
- Deprecated: A service is deprecated when it's superseded by a newer version, or if there is a planned obsolescence in a future release. A service in this state is still supported and becomes obsolete in a later release.
  - Note: You must use the active version of the service. If you were previously using a service that has been deprecated, then migrate to the new, active version.
- Obsolete: A service is obsolete when it's no longer shipped or supported.

## Message Patterns: Explained

All operations exposed on a business object service have both synchronous and asynchronous message patterns defined. For conciseness, the service documentation includes the definition for the synchronous message pattern only. Both synchronous and asynchronous operations have the same functional behavior, and request and response payloads. Custom object services don't have corresponding asynchronous operations.

#### Naming Conventions and Examples

The naming convention for the asynchronous operation is:

- Operation name: Synchronous operation name appended with Async
- Callback name: Synchronous operation name appended with AsyncResponse



Using Help Topic Abstract Service as an example, if the name of the synchronous operation is getEntityList, the asynchronous operation name and callback name would be getEntityListAsync and getEntityListAsyncResponse.

## Managing Web Service Sample Payloads: Worked Example

This example demonstrates how to add and edit a sample payload XML for a web service operation. It also describes how to delete a custom sample payload XML.

## Adding a Sample Payload XML

Follow these steps to add a sample payload XML for the getEntityList operation of Help Topic Abstract Service:

- 1. From the Navigator menu, select Tools Developer Connect.
- 2. On the **Web Service** overview page, search for Help Topic Abstract Service, and click the service display name.
- 3. On the Summary page, click the Sample Payloads tab, and then click Add Sample Payload.
- 4. Select getEntityList operation from the list, and enter a brief description.
- 5. Enter the payload XML, and click **OK**.

## Editing a Sample Payload XML

The Developer Connect portal displays predefined and custom payloads of the web services. You can edit only the custom payloads. Follow these steps to edit a custom sample payload of the <code>getEntityList</code> operation of <code>Help Topic Abstract</code> Service:

- 1. On the **Web Service** overview page, select Help Topic Abstract Service.
- 2. On the Summary page of the web service, click the Sample Payloads tab and select getEntityList operation.
- 3. In the Edit Sample Payload dialog box, edit the payload XML and click OK.

## Deleting a Sample Payload XML

You can delete only custom payloads, and not predefined payloads. Follow these steps to delete a custom sample payload of the getEntityList Operation of Help Topic Abstract Service:

- 1. On the **Web Service** overview page, select Help Topic Abstract Service.
- 2. On the **Summary** page of the web service, click the **Sample Payloads** tab and select getEntityList operation.
- 3. Click the delete icon for the selected operation and click **OK**.

# Importing Value Set Values to Oracle Applications Cloud: Worked Example

This example demonstrates how to use the information in the Developer Connect portal to import value set values to Oracle Applications Cloud.

Suppose a fictional partner application wants to extend it with certain functionality available in another application. Instead of creating the required business objects and attributes in this application, you can import them using a web service. Before you proceed with the import process, enable the access permissions for the web service and review the information in the Developer Connect portal:

- Synchronize the web services information
- · Review the web service details
- Update the request payload

# Synchronizing Web Services Information

You must synchronize the Developer Connect portal with your Oracle Applications Cloud instance to get the latest web service information. Follow these steps:

1. From the Navigator menu, select **Tools - Developer Connect**.



#### 2. Click Synchronize.

Note: The Last Refreshed date indicates when the Developer Connect portal was last synchronized with your cloud instance.

## Searching And Reviewing Web Service Information

On the Web Services overview page, you can enter Manage File Import and Export in the Find field. You can also use Advanced Search option to find web service names that contain import, and select the relevant service from the search results.

To review the information:

- 1. On the Web Services overview page, select Manage File Import and Export Service. The Summary panel shows information such as the display name, business object that the service defines, life cycle status, QName, security policy used, and a brief overview of the service.
- 2. Click the Operations tab to view the operations supported by the web service.
- 3. Click the uploadFiletoucm operation and review the result parameter in the response payload. This parameter holds the file ID in the WebCenter Content repository from which the value set values are imported.
- **4.** Click the **valueSetValuesDataLoader** operation and review the **fileIdAtRepository** parameter in the request payload. This parameter holds the file ID in the WebCenter Content repository.
- 5. Click **WSDL File** link of the service to download the web service information.

### **Updating Request Payload**

<soap:Body>

<ns1:uploadFiletoUCM

To add a sample payload to import the value set values:

- 1. Click the Sample Payloads tab and then click **Add Sample Payload**.
- 2. Select uploadFiletoUCM from the operation name list.
- 3. Enter a brief description of the payload in the description text box.
- **4.** Add the payload to get the file ID from the WebCenter Content repository:

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

xmlns:ns1="http://xmlns.oracle.com/oracle/apps/fnd/applcore/webservices/types/"

```
ns2="http://xmlns.oracle.com/oracle/apps/fnd/applcore/webservices/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<ns1:document xsi:type="ns2:DocumentDetails">
<ns2:fileName>VS123.txt</ns2:fileName>
<ns2:contentType>plain/text</ns2:contentType> <ns2:content>

VmFsdWVTZXRDb2RlfeluzGVwZW5kzW50VmFsdWV8SW5kZXBlbmRlbnRWYWx1ZU51bWJlcnxJbmRlcGVuZGVudFZhbHVlRGF0ZXxJbmRlcGVuZGVuQX R0cmlidXRlNH
xDdXN0b21WYwx1ZUF0dHJpYnV0ZTV8Q3VzdG9tVmFsdWVBdHRyaWJldGU2fEN1c3RvbVZhbHVlQXR0cmlidXRlN3xDdXN0b21WYwx1ZUF0dHJpY
Hx 8fHx8fHx8fHw K
UkVMN19CN19WU19OVU1fSU5EfHx8fHwyMDAwfHx8fHxUZXN0aW5nIGZvciBCdWcgMTczNzU2ODR8WXwyMDExLTewLTAxfDIwMTItMTAtMzB8Mnx
8f Hx8fHx8fHx8fHx8fApSRUw3X012X1ZTX0RBVEVfSU5EfHx8fHx8fDIwMzAtMDEtMDJ8fHxUZXN0aW5nIGZvciBCdWcgMTczNzU2ODQtMXxzf
UV fREVQfDIwMzE t
MDMtMDEgMDE6MDA6MDAuMDB8fHwyMDMxLTAzLTAxIDAxOjAwOjAwLjAwfDIwMzEtMDMtMDEgMDE6MDA6MDAuMDB8fHwyMDMxLTAzLTAxIDAxOjA
</ns1:content>
<ns2:documentAccount>fin$/tax$/import$</ns2:documentAccount>
<ns2:documentTitle>VS</ns2:documentTitle>
```

5. Click OK.

</ns2:document>
</ns1:uploadFiletoUCM>

</soap:Body>
</soap:Envelope>

6. Select valueSetValuesDataLoader from the operation name list.



- 7. Enter a brief description of the payload in the description text box.
- 8. Add the payload to read the contents from the file and import the value set values:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
  <ns1:valueSetValuesDataLoader

xmlns:ns1="http://xmlns.oracle.com/oracle/apps/fnd/applcore/webservices/types/">
  <ns1:fileIdAtRepository>1234</ns1:fileIdAtRepository>
  </ns1:valueSetValuesDataLoader>
  </soap:Body>
  </soap:Envelope>
```

# Cross-Origin Resource Sharing

## **CORS: Explained**

Cross-Origin Resource Sharing (CORS) enables secure cross domain communication from a browser. You can configure CORS headers to enable a client application running in one domain to retrieve resources from another domain, using HTTP requests. By default, browser-based programming languages, such as JavaScript, can access content only from the same domain. CORS provides a mechanism to overcome this limitation and access resources from different domains.

To enable CORS in Oracle Applications Cloud, you must set profile option values for the CORS headers in the Setup and Maintenance work area. This table lists the supported CORS headers.

CORS Header	Purpose
Access-Control-Allow-Origin	Contains a comma-separated list of trusted origins that a client application can access resources from.
Access-Control-Max-Age	Specifies the duration of storing the results of a request in the preflight result cache.
Access-Control-Allow-Methods	Contains a comma-separated list of permitted HTTP methods in a request.
Access-Control-Allow-Headers	Contains a comma-separated list of permitted HTTP headers in a request.
Access-Control-Allow-Credentials	Specifies whether a client application can send user credentials with a request.

### Example

A client application retrieves resource X from server A, which runs the application logic. The client application then makes an HTTP request to retrieve resource Y from server B. To allow this cross-server request from the client application, you must configure the \*Access-Control-Allow-Origin\* header in server A. Otherwise, the request fails and displays an error message.

#### Related Topics

- Managing Profile Option Values for CORS Headers: Points to Consider
- Setting Profile Option Values: Procedure



# Managing Profile Option Values for CORS Headers: Points to Consider

You can set profile option values for the CORS headers using the Manage Administrator Profile Values task in the Setup and Maintenance work area.

#### **CORS Headers**

This table lists the CORS headers that you can set profile option values for.

CORS Header	Profile Option Name (Profile Option Code)	Profile Option Values
Access-Control-Allow-Origin	Allowed Domains (ORACLE. ADF. VIEW. ALLOWEDORIGINS)	<ul> <li>Valid values for allowed origins:</li> <li>URL of the specific origin, for example, http://www.mydomain.com</li> <li>Comma-separated list of origins, for example, http://www.mydomain.com, http://adc6160507.us.oracle.com, http://software.dzhuvinov.com</li> <li>* to allow access to resources from all origins</li> <li>Empty (no value set) to prevent access to resources from any origin</li> <li>Note: You must set a value for this header to enable CORS.</li> </ul>
Access-Control-Max-Age	CORS: Access-Control-Max-Age (CORS_ACCESS_CONTROL_MAX_AGE)	Default value for caching preflight request is 3600 seconds.
Access-Control-Allow-Methods	CORS: Access-Control-Allow-Methods (CORS_ ACCESS_ CONTROL_ ALLOW_ METHODS)	Default values for allowed methods are OPTIONS, HEAD, GET, POST, PUT, PATCH, DELETE.
Access-Control-Allow-Headers	CORS: Access-Control-Allow-Headers (CORS_ ACCESS_ CONTROL_ ALLOW_ HEADERS)	Default values for allowed headers are Accept, Accept-Encoding, Cache-Control, Content-MD5, Content-Type, If-Match, If-None-Match, Origin, User-Agent, X-HTTP-Method-Override, X-Requested-By.
		Note: You must include Authorization, with a comma as the delimiter, to the list of allowed headers. For example: Accept, Accept- Encoding, Cache-Control, Authorization
Access-Control-Allow-Credentials	CORS: Access-Control-Allow-Credentials (CORS_ ACCESS_ CONTROL_ ALLOW_ CREDENTIALS)	True to enable sending credentials with the request



CORS Header	Profile Option Name (Profile Option Code)	Profile Option Values
		<ul> <li>False, which is the default value, to disable sending credentials with the request</li> </ul>

#### Related Topics

Setting Profile Option Values: Procedure

# Viewing Details About Predefined Scheduled Processes: Procedure

To use web services to run predefined scheduled processes, you need details about the processes. View job definitions that the processes are based on, for example to get information about parameters. You might also need to find security requirements for running the scheduled process.

#### **Job Definitions**

A job definition contains the metadata that determines how a scheduled process works and what options are available during submission.

To view job definitions:

- 1. Go to the Setup and Maintenance work area.
- 2. Run a search with **Manage Custom Enterprise Scheduler Jobs** as the search term.
- 3. In the search results, open the Manage Custom Enterprise Scheduler Jobs task for the application that contains the job definition. Tasks with names that end in **and Related Applications** include multiple applications.
- 4. In the Manage Job Definitions tab, select your job definition and click **Edit**.
  - Note: Predefined job definitions are marked with an asterisk.
- **5.** Cancel after you get the information you need.

## Security

Privileges provide the access needed to run specific scheduled processes. Privileges are granted to duty roles, which are granted to job roles. To see which job roles inherit the needed privileges, use the Security Console or the security reference manuals for the appropriate product family.

#### Related Topics

How can I see which applications a Manage Custom Enterprise Scheduler Jobs task includes?

# Files for Import and Export

# Files for Import and Export: Explained

You can import data into or export data out of the applications. A repository stores the content and the import and export processes handle the data movement into and out of the repository. Integration specialists stage data for import and export.



Application administrators run processes to import data in repositories of content to application transaction tables, or retrieve data exported from applications.

Aspects of managing files for import and export involve the following:

- Using the File Import and Export page
- Interacting with content management
- Uploading to facilitate import
- Downloading to facilitate export
- Determining the file size

## The File Import and Export Page

Use the File Import and Export page to upload content to or download content from the document repository of Oracle WebCenter Content. Search criteria on the page are limited to the minimum metadata of content management records needed for file import and export. To open the page, from the **Navigator** menu in the global area, select **Tools - File Import and Export**.

Contact the WebCenter Content Administrator for the following additional requirements:

- Information or assistance regarding general access to content management (including all metadata)
- Creating and managing accounts
- Programmatically uploading and downloading content

## Interacting with Content Management

Each user with access to the File Import and Export page is assigned to one or more accounts in Oracle WebCenter Content. Accounts organize and secure access to the content items.

# Uploading to Facilitate Import

Uploading a file creates a record in Oracle WebCenter Content. When you upload a file, you must also specify an account to which you upload the file. The account you specify determines which import process picks up that file to import it. You can upload any compatible file format, such as MIME, which the content repository can parse. However, the uploaded format must conform to the requirements of the import process. For example, the comma-separated values (CSV) file for the Load Interface File for Import process.

# Downloading to Facilitate Export

Records in the search results table of the File Import and Export page provide download links to the files.

#### File Size

Upload and download don't apply the following by default:

- Data compression
- File splitting

The UPLOAD\_MAX\_DISK\_SPACE parameter in the web.xml file determines the maximum allowable file size in content management. The default maximum size is 10240000 (10MB).



# Files for Import and Export: Points to Consider

Interaction between the File Import and Export page and Oracle WebCenter Content requires securing content in an account. You can use the predefined accounts that are available in Oracle WebCenter Content.

Areas of file import and export involve the following:

- Defining security
- · Searching records
- · Accessing content in a new account
- Naming the account
- Deleting files

## **Defining Security**

You require the File Import and Export Management duty role for accessing the File Import and Export page. This duty role is included in the predefined role hierarchy for integration specialist roles and product family administrator roles. Files in Oracle WebCenter Content are associated with an account so that only users having access to that account can work with those files. Account names are unique and each account is treated as discrete by access control. You can only upload and download files to and from content repositories that are linked to the accounts you can access. The underlying integrated content management handles security tasks such as virus scanning.

## Searching Records

A record in Oracle WebCenter Content contains the metadata used for accessing the file. When a scheduled process is run on a file, the record for the file is assigned a process ID.

# Accessing Content in a New Account

After you create a new account in Oracle WebCenter Content, restart the content server. Otherwise, when you use the File Import and Export page to access content in the new account, you may experience a delay. That's because the policy store is being updated with the new account information.

# Naming the Account

If you create custom accounts for importing or exporting data, use the following conventions for naming the account:

- Don't include a forward slash (/) at the beginning or end.
- End the name with a dollar symbol (\$) to avoid partial string matching.
- Use (\$/) as a separator in the hierarchical structure.

For example: fin\$/journal\$/import\$ The File Import and Export page transforms account names by removing the \$ separators. For example fin\$/journal\$/import\$ appears as fin/journal/import. The Remote Introdoc Client (RIDC) HTTP command-line interface (CLI) transforms the account name you specify without the \$ symbol to one that includes the symbol. For example, fin/journal/import becomes fin\$/journal\$/import\$ in WebCenter Content.

# **Deleting Files**

You can delete one file at a time when you use the File Import and Export page. To delete multiple files simultaneously from the content repository, use the standard service page in Oracle WebCenter Content.



# External Data Integration Services for Oracle Cloud

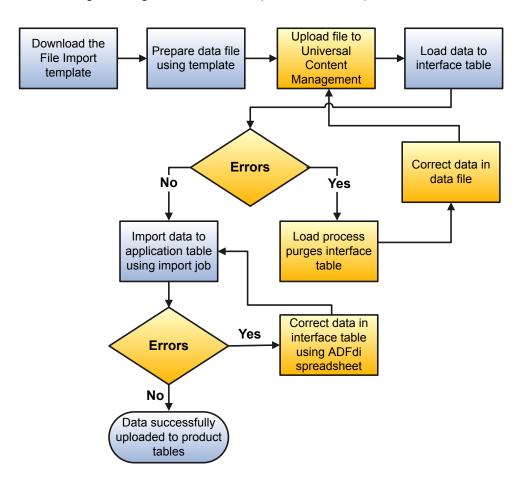
# External Data Integration Services for Oracle Cloud: Overview

Use External Data Integration Services for Oracle Cloud to load data into Oracle Fusion Applications from external sources, such as legacy systems and third-party applications.

External Data Integration Services for Oracle Cloud include the following components:

- Templates to structure, format, and generate the data file according to the requirements of the target application tables.
- File-based load process to load the data files into the interface tables.
- Application-specific data import processes to transfer data from interface tables to the application tables in your Oracle Fusion Applications.

The following flow diagram outlines the steps involved in the process:



For further information, see Using External Data Integration Services for Oracle ERP Cloud (2102800.1) on My Oracle Support at https://support.oracle.com.



#### Related Topics

Using External Data Integration Services for Oracle ERP Cloud

# Locating File Import Templates: Explained

The File Based Data Import guides in the Oracle Help Center (http://docs.oracle.com) include integration templates to help you prepare external data for loading and importing. Each template includes table-specific instructions, guidelines, formatted spreadsheets, and best practices for preparing the data file for upload. Use the templates to ensure that your data conforms to the structure and format of the target application tables.

Preparing external data using templates involve the following tasks:

- Downloading templates
- Preparing data using the XLS template

## **Downloading Templates**

To download the templates:

- 1. Open the File Based Data Import guide for your cloud service.
- 2. Locate the import process.
- 3. View the list of files.
  - Control files describe the logical flow of the data load process.
  - XLSM templates include the worksheets and macros for structuring, formatting, and generating your data file.
    - Note: You can use XML templates to import data into Oracle Data Integrator.
- 4. Click the template link in the File Links table to download the file. For example, click **JournalImportTemplate.xlsm** in the Journal Import topic.

# Preparing Data Using the XLS Template

To prepare your data in a spreadsheet format:

- 1. Open the XLS template. The first worksheet in each file provides instructions for using the template.
  - Note: If you don't follow the instructions, you'll get data load errors and data import failures.
- 2. Save the file.
- 3. Click the **Generate CSV File** button.

The macro generates a comma-separated values (CSV) file and compresses the file into a ZIP file. You must transfer the ZIP file to the content management server.

# Opening the XML Template

To prepare your data in Oracle Data Integrator, download the XML templates using the following steps:

1. Import the family-level template as a model folder.



- 2. Import the product-level template as a model folder within the family-level model folder.
- 3. Import the product template as a model within the product-level model folder.
- 4. Create the integration project.
- 5. Create the package.
- 6. Add and configure these elements:
  - Integration projects
  - Content management document transfer utility
- 7. Execute the package. The package generates the CSV file and compresses it into a ZIP file.

# Using Excel Integration Templates to Generate Data Files: Points to Consider

The File Based Data Import guides in the Oracle Help Center (http://docs.oracle.com) include integration templates to help you prepare external data for loading and importing. Each template includes table-specific instructions, guidelines, formatted spreadsheets, and best practices for preparing the data file for upload. Use the templates to ensure that your data conforms to the structure and format of the target application tables.

### Template Structure

The integration templates include the following characteristics:

- Each interface table is represented by a separate worksheet.
- Each interface table field is represented by a worksheet column with a header in the first row.
- Each column header contains bubble text or comments that include details about the column, such as the expected data type, length, and, in some cases, other instructional text.
- Columns are formatted, where applicable, to match the target field data type to eliminate data entry errors.

The worksheet columns appear in the order that the control file processes the data file.

For more information on the template structure, see the Instructions and CSV Generation worksheet in the template.

# Template Requirements

To minimize the risks of an unsuccessful data load, ensure the following:

- Unused columns can be hidden, but not reordered or deleted.
  - ▲ Caution: Deleting or reordering columns causes the load process to fail and results in an unsuccessful data load.
- External data must conform to the data type accepted by the control file and process for the associated database column.
- Date column values must appear in the YYYY/MM/DD format.
- Amount column values can't have separators other than a period (.) as the decimal separator.
- Negative values must be preceded by the minus (-) sign.
- Column values that require whole numbers include data validation to allow whole numbers only.



• For columns that require internal ID values, refer to the bubble text for additional guidance about finding these values.

After you finish preparing the data in the sheet, click the **Generate CSV File** button to generate a ZIP file containing one or more CSV files.

# Using XML Templates to Generate Data Files for Integration: Explained

The File Based Data Import guides in the Oracle Help Center (https://docs.oracle.com) include XML integration templates that you use with Oracle Data Integrator to generate import files from your external data. Oracle Data Integrator provides a solution for integrating complex data from a variety of sources into your Oracle Fusion applications.

To use the XML templates and generate the import files, you must:

- Install and set up Oracle Data Integrator
- Create source and target models
- Create integration projects
- Note: For Oracle Cloud implementations, you must upload the ZIP file to the content management repository in Oracle Cloud. For non-Cloud implementations, you can streamline the data integration process by installing the content management document transfer utility, which uses Oracle Data Integrator to transfer the ZIP file.

# Using XML Integration Templates to Generate Data Files: Points to Consider

Use XML templates in Oracle Data Integrator to prepare your external data for the load and import process.

The File Based Data Import guides in the Oracle Help Center (https://docs.oracle.com) include three types of XML templates that you import as target models in your Oracle Data Integrator repository:

- Family level
- Product level
- Product

## Family-Level XML Files

A family-level XML file is common to a group of product-level model folders and product models.

Consider the following points when you use family-level XML files:

- Use the family-level XML file to support assets in the family, for example, Oracle Fusion Financials or Human Capital Management.
- Import the family-level XML file into your Oracle Data Integrator repository prior to importing the other XML files.
- Import one family-level XML file as a model folder for each family of products.
- Import each family-level XML file as a top-level model folder.
- Import the family-level XML file one time; it supports all subsumed product-level model folders.



Select Synonym Mode Insert Update as the import type.

#### Product-Level XML Files

A product-level XML file is common to a group of product models.

Consider the following points when you use product-level XML files:

- Use the product-level XML file to support assets in the product line, for example, Fixed Assets, General Ledger, or Payables.
- Import one product-level XML file as a model folder for each line of products.
- Import the product-level XML file as a model folder into your Oracle Data Integrator repository.
- Import the family-level XML file before you import product XML files.
- Import each product-level XML file as a mid-level model folder within the appropriate family-level model folder.
- Import the product-level XML file one time; it supports all subsumed product models.
- Select Synonym Mode Insert Update as the import type.

#### **Product XML Files**

A product XML file represents a specific interface table asset.

Consider the following points when you use product XML files:

- Import one product XML file as a model for each interface table or set of tables, for example, Mass Additions.
- Import the product XML file as a model into your Oracle Data Integrator repository after you import the product-level XML file.
- Import each product XML file as a model within the appropriate product-level model folder.
- Import each product XML file one time. The model is based on File technology.
- Select Synonym Mode Insert Update as the import type.
- After you import the product model, connect the model to the correct logical schema.

# Creating Integration Projects That Generate Data Files for Import: Points to Consider

When you use Oracle Data Integrator (ODI) to generate the import data files from external data sources, you must configure an integration project. Integration projects are collections of ODI components that provide the procedural details of an integration from a source to a target. The source is your external data and the target is the import data file that you load and import into your Oracle Fusion Applications.

To create your integration project, you configure the following components:

- Knowledge modules
- Integration interfaces

## Knowledge Modules

Knowledge modules contain the information that Oracle Data Integrator requires to perform a specific set of tasks against a specific technology or set of technologies. For example, check knowledge modules ensure that constraints on the sources and targets are not violated, and integration knowledge modules load data to the target tables.



Consider the following points about knowledge modules:

- Knowledge modules that you import into your integration project depend on the source and target technologies, as well as other integration-specific rules and processes.
- Multiple types of knowledge modules exist in ODI.
- Use the SQL File to Append module to create the import data file.

### Integration Interfaces

Integration interfaces contain the sets of rules that define the loading of data from one or more sources to the target.

Consider the following points about integration interfaces:

- The source is the data store from your external data model.
- The target is the interface table data store, which is the CSV file from your interface table model.
- After you set up the source and target data stores, map the target fields to the source fields, or map source field values to target fields or constants.

# Transferring Data Files to Oracle WebCenter Content Using Manual Flow: Explained

After you generate the ZIP file that contains the CSV data import file, transfer the ZIP file to the content repository.

Use any of the following methods to transfer file:

- · File Import and Export page in Oracle Fusion Applications: Manual flow
- Oracle Fusion ERP Integration web service: Automated flow

Aspects of transferring data files to content management involve the following:

- Target accounts
- Accessing transferred content

# Predefined Target UCM Accounts

You can transfer data files to predefined accounts in the Universal Content Management server that correspond to the interface table or assets.

To find the UCM account:

- 1. Open the File Based Data Import guide for your cloud service.
- 2. Locate your respective import process. For example, **Journal Import**.
- 3. View the UCM account in the Details section.

For more information, see the following guides in the Oracle Help Center (https://docs.oracle.com):

- SOAP Web Services guide for your cloud services
- File Based Data Import guide for your cloud services



# Document Transfer Utility: Explained

The WebCenter Content Document Transfer Utility for Oracle Fusion Applications is a feature-set Java library that provides programmatic access to the content repository. Use the utility to import and export documents, such as import files that contain external data that you want to load into interface and application tables.

The library includes:

- Oracle WebCenter Content client command line tool
- Oracle Data Integrator (ODI) upload and download tools
- Oracle WebCenter Content remote intradoc client (RIDC)
- Oracle HTTPClient
- Oracle Fusion Applications branding and defaults

Options for the WebCenter Content Document Transfer Utility for Oracle Fusion Applications fall into these categories:

- DownloadTool program options
- UploadTool program options
- Debugging and silent invocation options

## DownloadTool Program Options

This table describes the download tool program options:

Option	Description
url	Protocol-specific connection URL of content server
username	Username to leverage
password	Password, supplied in command line
passwordFile	Password, supplied in text file on the first line of the file
dID	ID of document revision to download
	dID is unique across repository
	dID changes with each revision
	Note: Alternatively, specify the dDocName and RevisionSelectionMethod to identify the dID to leverage.
dDocName	Content name
	Multiple revisions of a document can share the same dDocName value, otherwise it is unique.



Option	Description	
	Note: You should also provide RevisionSelectionMethod value.	
RevisionSelectionMethod	Revision to download	
	Valid values: Latest, LatestReleased	
	Default value: Latest	
outputFile	Path and name of local file to write	

Here you see a sample download invocation command:

java -classpath "oracle.ucm.fa\_client\_11.1.1.jar" oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic password=welcom3i
dID=21537 outputFile="/tmp/output.doc"

Here you see sample output:

Oracle WebCenter Content Document Transfer Utility Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing download (GET\_FILE) ...
Download successful.

## UploadTool Program Options

This table describes the upload tool program options:

Option	Description
url	Protocol-specific connection URL of content server
username	Username to leverage
password	Password, supplied in command-line
passwordFile	Password, supplied in text file on the first line of the file
primaryFile	Fully-qualified path of local primary file to upload
dDocAccount	Destination account
dDocTitle	Document title
checkout	If uploading a document revision, check out the document from the repository before uploading the revision
	Valid values: true, false
	Default value: false



Option	Description
ignoreCheckoutErrorNeg22	Ignore error -22 (user has already checked-out the document) when checking-out the document.
	Valid values: true, false
	Default value: true

Here you see a sample upload invocation command:

```
java -classpath "oracle.ucm.fa_client_11.1.1.jar" oracle.ucm.client.UploadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic password=welcom3i
primaryFile="/tmp/resume.doc" dDocTitle="Resume of MSMITH" -dDocAccount=/acme/sales
```

Here you see sample output:

```
Oracle WebCenter Content Document Transfer Utility
Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing upload (CHECKIN_UNIVERSAL) ...
Upload successful.
[dID=21537 | dDocName=UCMFA021487]
```

# Debugging and Silent Invocation Options

This table describes the usable options which are common to all tools.

Option	Description
verbose	Verbose output
	Log filled with Request/ Response DataBinders
quiet	Minimal output
version	Print tool revision or version
log_file_name	Send program output to specified log file instead of the System.out log file
log_file_append	Append log to existing log file rather than overwrite it
	Valid values: true, false
	Default value: false
socketTimeout	Override time out of socket
	Specify override time in seconds

You can use the tools to test the connection. Provide only the url, username, and password as you see in this sample test:

java -classpath "oracle.ucm.fa\_client\_11.1.1.jar" oracle.ucm.client.DownloadTool
url=http://ucmserver.com:16200/cs/idcplg username=weblogic password=welcom3i

Here you see the sample output:



Oracle WebCenter Content Document Transfer Utility Oracle Fusion Applications
Copyright (c) 2013, Oracle. All rights reserved.
Performing connection test (PING\_SERVER) ...
Connection test successful.

# Load Interface File for Import Process

Use to load external setup or transaction data from a data file in the content repository to interface tables. The process prepares the data for import into application tables.

You run this process from the Scheduled Processes page. You can run it on a recurring basis.

Before running this process, you must:

- 1. Prepare your data file.
- 2. Transfer the data file to the content repository.

#### **Parameters**

#### **Import Process**

Select the target import process.

#### Data file

Enter the relative path and the file name of the \*.zip data file in the content repository.

# Importing Data into Application Tables: Procedure

The final destination for your external data is the application data tables of your Oracle Fusion application.

Importing data into application tables involves the following:

- Loading data into interface tables
- Finding and submitting the import process

## Loading Data into Interface Tables

Interface tables are intermediary tables that store your data temporarily while the application validates format and structure. Run the Load Interface File for Import scheduled process to load data from the data file into the interface table that corresponds to the template that you use to prepare the data.

To load your data into interface tables, submit the Load Interface File for Import scheduled process using the following steps:

- 1. From the Navigator, click **Tools**.
- Click Scheduled Processes.
- 3. Click the Schedule New Process button.
- 4. Search and select the Load Interface File for Import job.
- 5. On the Process Details page:
  - a. Select the target import process.
  - **b.** Enter the data file name.



6. Submit the process.

If the process is successful, the status is SUCCEEDED and the process populates the interface tables. If the process isn't successful, the status is ERROR.

Note: The data file remains in the content repository after the process ends.

## Finding and Submitting the Import Process

Run the appropriate import process to import the data into the interface tables of your Oracle Fusion application.

To import your data into the application tables:

- 1. From the Navigator, click **Tools**.
- 2. Click Scheduled Processes.
- 3. Click the Schedule New Process button.
- 4. Search and select the import process for the target application tables.
- **5.** On the Process Details page, select the process that corresponds to the data that you're importing. For example, **Journal Import**.

If you prepared your data using the spreadsheet template, select the process shown in the Overview section of the spreadsheet.

6. Submit the process.

If the process is successful, the status is SUCCEEDED. The data in the interface tables is validated and the successful records are imported into the Oracle Fusion application tables. If the process isn't successful, the status is ERROR.

Note: For more information on the process used for data prepared using the spreadsheet template, see the Instructions and CSV Generation tab of the spreadsheet template.

# Correcting Import Load Process Errors: Explained

The Load Interface File for Import process ends in error if the load of the data file fails on any row. The Load File to Interface child process ends as an error or warning. All rows that were loaded by the process are deleted and the entire batch of records is rejected.

# Correcting Data Upload Errors

To correct errors:

- 1. Review the error logs.
- 2. Change any structural or formatting anomalies in the data.
- 3. Generate the ZIP file containing the CSV files using the template.
- 4. Upload the file to the UCM server and resubmit the Load Interface File for Import process.
- 5. Repeat these steps until the process successfully loads all the data.

# Correcting Import Process Errors

If the import process fails with errors:

- 1. Review the errors in the import log.
- 2. Correct the error records using the ADFdi correction spreadsheets.



# 11 Importing and Exporting Setup Data

# Configuration Packages: Explained

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 submit export, a snapshot of the appropriate setup data is added to the configuration package using the definition. You can continue making modifications to the setup data in the environment and create a new configuration package any time you need it.

You can generate the setup export and import definition implicitly or explicitly:

- A configuration package is created implicitly when you export setup data for an entire offering or any functional area from the Applications Administration page.
- A configuration package is created explicitly when you export setup data based on an implementation project. This
  method enables further customization of the configuration packages.

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 determines the export and import sequence.

The tasks and their associated business objects in the selected configuration (offering, functional area or implementation project) define the setup export and import definition for the configuration package. In addition, the sequence of the tasks in the implementation project determines the export and import sequence.

Once a configuration package is exported, the setup export and import definition is locked and cannot be changed; that is, you cannot change the selection (add or remove) of tasks and their associated business objects, change their export and import sequence, nor the scope value selection. However, you can create a new configuration package with such modifications at any time.

# Implementation Project Based Export and Import: Explained

Use an implementation project as the source for exporting setup data when you are required to customize the list of tasks or of objects you want to export setup data for.

You must explicitly create a configuration package from the Manage Configuration Packages page to export setup data for an implementation project. 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. Depending on your needs, when you create a configuration package based on an implementation project, you can also customize some additional aspects, as explained here.

Exclude some of the business objects from the configuration you selected to export setup data for.



You should limit to use this option when the corresponding setup data is already available in the target instance and therefore no data dependency issues appear during import.

Change the default import sequence of the business objects

Change the default import sequence of the business objects. It's strongly recommended that you limit using this option when you need to correct a data dependency issue and you fully understand the data relationships between the business objects of your configuration.

Filter the setup data to export

## **Export**

During export, appropriate setup data is identified based on the tasks in the implementation project used as source for 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. Once export completes, you can download the configuration package file as a zipped archive of multiple XML files, move it to the target application instance, and upload and import it. After exporting the setup data you may continue entering new or modifying existing setup data for your configuration. Since the configuration package is a snapshot of the setup data taken at the time export is initiated, you may need to take another snapshot of the same configuration or set of data later. Although you can always create a different configuration package, FSM provides you the ability to take another snapshot of the setup data using the same customized export and import definition by exporting the configuration package multiple times and 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**

During import, you first upload a configuration package created by the export process and then import the setup data. All setup data contained in the configuration package is imported into the environment you initiate the setup data import from. In the target application instance, the setup import process inserts 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.

# Moving Common Reference Objects

## Moving Common Reference Objects: Overview

The common reference objects 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, moving from test to the production phase of an implementation, attend 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 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, you must note all the dependencies before moving the objects so that there are no broken references among them.

# Business Objects for Moving Common Reference Objects: Points to Consider

Common reference objects in Oracle Fusion Functional Setup Manager are used to move application setup content from one environment to another. For example, from a test environment to a production environment.

#### Choice of Parameters

The following table lists the business objects, the movement details, and the effect of the setup task parameter on the scope of the movement.

#### Note:

- You can move only the translations in the current user language.
- You can move the Oracle Social Network business objects and the Navigator menu customizations using the customization sets on the Customization Migration page.

Business Object Name	Moved Functional Item	Effect on the Scope of Movement
Application Message	Messages and associated tokens	No parameters: All messages are moved.
		Parameter moduleType/ moduleKey Only messages belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter 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.
		Parameter 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.



Business Object Name	Moved Functional Item	Effect on the Scope of Movement
		Parameter moduleType/ moduleKey Only attachment categories belonging to the specified module and its descendant modules in the taxonomy hierarchy along with the respective category-to-entity mappings are moved.
Application Document Sequence Category	Document sequence categories	No parameters: All categories are moved.
		Parameter moduleType/ moduleKey Only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter code/ applicationId Only the specified document sequence category code is moved.
Application Document Sequence	Document sequences and their assignments	No parameters: All sequences are moved.
		Parameter moduleType/ moduleKey Only document sequences belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved
		Parameter name: Only the specified document sequence is moved.
Application Descriptive Flexfield	Descriptive flexfield registration data and setup data	No parameters: All descriptive flexfields are moved.
		Parameter moduleType/ moduleKey Only descriptive flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter descriptiveFlexfieldCode/ applicationId Only the specified descriptive flexfield is moved. Importing the metadata of a flexfield can change its deployment status. Therefore, you must redeploy if there are any affected flexfields. The import process automatically submits affected flexfields for redeployment. Also only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.
Application Extensible Flexfield	Extensible flexfield registration data and setup data, including categories	No parameters: All extensible flexfields are moved
		Parameter moduleType/ moduleKey Only extensible flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter extensibleFlexfieldCode/ applicationId Only the specified extensible flexfield is moved. Importing the metadata of



Business Object Name	Moved Functional Item	Effect on the Scope of Movement
		a flexfield can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment.
		Also, only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.
Application Key Flexfield	Key flexfield registration data and setup data	No parameters: All key flexfields are moved.
		Parameter moduleType/ moduleKey Only key flexfields belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter keyFlexfieldCode/ applicationId Only the specified key flexfield is moved.
		Importing the metadata of a flexfield can change its deployment status and therefore, the affected flexfields must be redeployed. The import process automatically submits affected flexfields for redeployment. Only flexfields with a deployment status of Deployed or Deployed to Sandbox are eligible to be moved.
Application Flexfield Value Set	Value set setup data	No parameters: All value sets are moved.
		Parameter moduleType/ moduleKey Only value sets belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter valueSetCode: Only the specified value set is moved.
		Importing the metadata of a value set can change the deployment status of flexfields that use the value set. Therefore, you must redeploy if there are any affected flexfields. The import process automatically submits affected flexfields for redeployment.
Application Reference Currency	Currency data	No parameters: All currencies are moved.
Application Reference ISO Language	ISO language data	No parameters: All ISO languages are moved.
Application Reference Industry	Industry data including industries in territories data	No parameters: All industries are moved.
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.



Business Object Name	Moved Functional Item	Effect on the Scope of Movement
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	No parameters: All standard lookups are moved.
		Parameter moduleType/ moduleKey Only standard lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter lookupType: Only the specified common lookup is moved.
Application Common Lookup	Common lookup types and their lookup codes	No parameters: All common lookups are moved.
		Parameter moduleType/ moduleKey Only common lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter lookupType: Only the specified common lookup is moved.
Application Set-Enabled Lookup	Set-enabled lookup types and their lookup codes	No parameters: All set-enabled lookups are moved.
		Parameter moduleType/ moduleKey Only set-enabled lookups belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		Parameter lookupType: Only the specified set-enabled lookup is moved.
Application Profile Category	Profile categories	No parameters: All profile categories are moved.
		Parameter moduleType/ moduleKey Only categories belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.
		name/ applicationId Only the specified category is moved.
Application Profile Option	Profile options and their values	No parameters: All profile options and their values are moved.
		Parameter moduleType/ moduleKey Only profile options and their values belonging to the specified module are moved.



Business Object Name	Moved Functional Item	Effect on the Scope of Movement
		Parameter profileOptionName: Only the specified profile option and its values are moved.
Application Profile Value	Profile options and their values	No parameters: All profiles and their values are moved.
		Parameter moduleType/ moduleKey Only profiles and their values belonging to the specified module are moved.
		Parameter categoryName/ categoryApplicationId Only profiles and their values belonging to the specified category are moved.
		Parameter profileOptionName: Only the specified profile and its values are moved.
Application Reference Data Set	Reference data sets	No parameters: All sets are moved.
Application Reference Data Set Assignment	Reference data set assignments	Parameter determinantType: Only assignments for the specified determinant type are moved.
		Parameter determinantType/ referenceGroupName Only assignments for the specified determinant type and reference group are moved.
Application Tree Structure	Tree structures and any labels assigned to the tree structure	No parameters: All tree structures (and their labels) are moved.
		Parameter moduleType/ moduleKey Only tree structures (and their labels) belonging to the specified module are moved.
		Parameter treeStructureCode: Only the specified tree structure (with its labels) is moved.
Application Tree	Tree codes and versions	No parameters: All trees are moved.
		Parameter moduleType/ moduleKey Only trees belonging to the specified module are moved.
		Parameter treeStructureCode: Only trees belonging to the specified tree structure are moved.
		Parameter TreeStructureCode/ TreeCode Only trees belonging to the specified tree structure and tree code are moved.
Application Tree Label	Tree structures and any labels assigned to the tree structure	No parameters: All tree structures (and their labels) are moved.



Business Object Name	Moved Functional Item	Effect on the Scope of Movement  Parameter moduleType/ moduleKey Only tree structures (and their labels) belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  Parameter treeStructureCode: Only the specified tree structure (with its labels) is moved.
Application Data Security Policy	Database resources, actions, conditions, and data security policies	No parameters: All database resources/actions/ conditions/ policies are moved.  Parameter moduleType/ moduleKey Only database resources/ actions/ conditions/ policies belonging to the specified module and its descendant modules in the taxonomy hierarchy are moved.  Parameter objName: Only the specified database resource along with its actions/ conditions/ policies is moved.  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.

# 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, ensure that these dependencies or references aren't broken or lost.

### 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 decide 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 decide to move only value sets, or move both value sets and their lookups as part of the same package. Whatever be the combination, Oracle recommends 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 following order:

1. Move created taxonomy modules before moving any objects that reference them, such as flexfields, lookups, profiles, messages, and so on.



- 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.
- 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 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 initiate 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

#### accounting flexfield

The structure that determines the chart of accounts, including the number and order of the individual segments, as well as assigning the value sets to the segments.

#### action

The kind of access, such as view or edit, named in a security policy.

#### **ADF**

Acronym for Application Developer Framework. A set of programming principles and rules for developing software applications.

#### analytics

Business intelligence objects such as analyses and dashboards that provide meaningful data to help with decision making.

#### 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.

#### 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.

#### autosuggest

Suggestions that automatically appear for a search field, even before you finish typing your search term. You can select any of the suggestions to run your search.

#### balancing segment

A chart of accounts segment used to automatically balance all journal entries for each value of this segment.

#### business function

A business process or an activity that can be performed by people working within a business unit. Describes how a business unit is used.

#### business intelligence catalog

The repository where all business intelligence objects, including analytics, reports, briefing books, and agents, are stored. The catalog contains separate folders for personal, shared, and custom objects.



#### 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.

#### clause numbering level

Specifies the determinant type of the document sequence for automatic clause numbering

#### condition

The part of 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 be associated with a different set of context-sensitive segments.

#### context-sensitive segment

A flexfield segment that may or may not appear depending upon a context. Context-sensitive segments are custom attributes that apply to certain entity rows based on the value of the context segment.

#### 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 that lets you create and evaluate business rules in the terms library such that the contract terms meet your business standards, by suggesting contract changes or additional clauses.

#### **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.

#### **CORS**

Acronym for Cross-Origin Resource Sharing. A web service standard to enable a client application running in one domain to retrieve resources from another domain, using HTTP requests.

#### cost center

A unit of activity or a group of employees used to assign costs for accounting purposes.

#### 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 security

The control of access and action a user can take 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 a customer case. You may configure information collection and storage based on the context.



#### determinant

A value that specifies the use of a reference data set in a particular business context.

#### determinant type

An optional value that affects document sequencing in a transaction. The available determinant types are Business Unit, Ledger, Legal Entity, and Tax Registration.

#### determinant type

The value that affects sharing of reference data in a transaction across organizations, such as a business unit or a cost organization.

#### determinant value

A value specific to the selected determinant type of a document sequence. 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. It is relevant in a document sequence assignment only if the document sequence has a determinant type.

#### 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.

#### enterprise

An organization with one or more legal entities under common control.

#### entitlement

Grant of access to functions and data. Oracle Fusion Middleware term for privilege.

#### extensible flexfield

Customizable expansion space used to capture multiple sets of information within a context or multiple contexts. Some extensible flexfields let you group contexts into categories.

#### external system or external application

A system or application that is external to and not part of Order Management. An order capture system that resides upstream of Order Management is an example of an external system. A fulfillment application that resides downstream of Order Management is an example of an external application.



#### 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

A flexible data field that you can customize to contain one or more segments or store additional information. Each segment has a value and a meaning.

#### flexfield segment

An extensible data field that represents an attribute and captures a value corresponding to a predefined, single extension column in the database. A segment appears globally or based on a context of other captured information.

#### global area

The region at the very top of the user interface that remains the same no matter which page you're on.

#### global search

The search in the global area that lets you search across many business objects.

#### grade

A component of the employment model that defines the level of compensation for a worker.

#### import

In the context of data integration, the transfer of data from interface tables to **application tables**, where the data is available to application users.

#### incident

A collection of diagnostic information about a critical error, providing details about the state of the application when the issue occurred.

#### interface table

A database table that stores data during data transfer between applications or from an external system or data file.

#### item master

A collection of data that describes items and their attributes recorded in a database file.

#### 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 definition

The metadata that determines what a job does and what options are available to users when they submit the scheduled process. A job is the executable for a scheduled process.

#### job role

A role, such as an accounts payable manager or application implementation consultant, that usually identifies and aggregates the duties or responsibilities that make up the job.

#### key flexfield

Configurable flexfield comprising multiple parts or segments, each of which has a meaning either individually or in combination with other segments. Examples of key flexfields are part numbers, asset category, and accounts in the chart of accounts.

#### 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, you can define multiple structures for a single key flexfield.

#### key flexfield structure instance

An occurrence of a key flexfield structure that shares the same order of segments as other instances of the key flexfield structure. However, each instance uses different value sets to validate the segments.

#### legal authority

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

#### legal employer

A legal entity that employs people.

#### legal entity

An entity identified and given rights and responsibilities under commercial law through the registration with country's appropriate authority.

#### legal jurisdiction

A physical territory, such as a group of countries, single country, state, county, parish, or city, which comes under the purview of a legal authority.

#### legal reporting unit

The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.



#### legislative data group

A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.

#### line of business

Set of one or more highly related products which service a particular customer transaction or business need. Refers to an internal corporate business unit.

#### load

In the context of data integration, the transfer of external data from data files to the receiving **interface tables** in preparation for an import into application tables.

#### lookup code

An option available within a lookup type, such as the lookup code BLUE within the lookup type COLORS.

#### lookup type

The label for a static list that has lookup codes as its values.

#### mainline metadata

The primary branch of metadata that a sandbox is published to. Once published, changes made in the sandbox become available to all users.

#### offering

A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.

#### party fiscal classification

A classification used by a tax authority to categorize a party for a tax.

#### payroll statutory unit

A legal entity registered to report payroll tax and social insurance. A legal employer can also be a payroll statutory unit, but a payroll statutory unit can represent multiple legal employers.

#### position

A specific occurrence of one job that is fixed within one department. It is also often restricted to 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 of access to functions and data; a single, real world action on a single business object.

#### profile option

User preferences and system configuration options that users can configure to control application behavior at different levels of an enterprise.

#### profile option level

The category or layer that defines a profile option. Site, Product, and User are the predefined levels.

#### profile option value

The setting mapped to the level of a profile option. A profile option may have multiple values set at different levels, such as Site or User.

#### **Query By Example**

The row of fields directly above table column headers, used for filtering the data in the table.

#### reference data

Data in application tables that is not transactional or high-volume, which an enterprise can share across multiple organizations. For example, sales methods, transaction types, or payment terms.

#### 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 collection of reference data sets that correspond to logical entities, such as payment terms defined across multiple tables or views. Based on the common partitioning requirements across entities, the reference data sets are grouped to facilitate data sharing among them.

#### registration

The record of a party's identity related details with the appropriate government or legal authorities for the purpose of claiming and ensuring legal and or commercial rights and responsibilities.



#### report

An output of select data in a predefined format that's optimized for printing.

#### role

Controls access to application functions and data.

#### sandbox

A testing environment that isolates untested code changes from the mainline environment so that these changes don't affect the mainline metadata or other sandboxes.

#### scheduled process

A program that you run to process data and, in some cases, generate output as a report.

#### segment

A segment is a single field within a flexfield and maps to a single table column in your database. When customizing a flexfield, you define the appearance and meaning of individual segments.

#### service provider model

A business unit that provides specific business functions for another business unit.

#### set

Classified and grouped reference data that organizational entities share.

#### set enabled

A property that describes entities that an organization shares as reference data. For example, you can indicate a lookup, customer, location, or document attachment as set enabled.

#### 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.



#### suggestion group

Category of suggestions that appear in the autosuggest for the global search.

#### tax registration

The registration of a party with a tax authority that confers tax rights and imposes certain tax obligations.

#### territory

A legally distinct region 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

A set of guidelines or a framework applied to create a tree, include data, version a tree, or access a tree.

#### tree version

An instance of a tree that includes life cycle elements such as start and end dates, and indicates whether the tree is active. If a tree is associated with a reference data set, all tree versions belong to one set.

#### user rate type

Rate you enter at journal entry time to convert foreign currency transactions to your ledger 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 area

A set of pages containing the tasks, searches, and other content you need to accomplish a business goal.

#### work relationship

An association between a person and a legal employer, where the worker type determines whether the relationship is a nonworker, contingent worker, or employee work relationship.

#### workflow

An automated process that passes a task from one user (or group of users) to another to view or act on. The task is routed in a logical sequence to achieve an end result.

