User Defined Fields User Guide

Oracle Banking Corporate Lending Release 12.5.0.0.0

Part Number E88408-01

September 2017

User Defined Fields User Guide
September 2017
Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India
Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax:+91 22 6718 3001
www.oracle.com/financialservices/

Contents

1. Preface

1.1 Introduction

This manual is intended as a guide to help you define User Defined Fields (UDFs) and associate the UDFs with products (and therefore on contracts involving the products) and on specific Functions in Oracle Lending. You can further obtain information specific to a particular field by placing the cursor on the relevant field and pressing <F1> on the keyboard.

1.2 Documentation Accessibility

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.

1.3 Organization

This manual is organized as follows:

Chapter	Description
Chapter 1	*About this Manual* gives information on the intended audience. It also lists the various chapters covered in this User Manual.
Chapter 2	*Creating Custom Fields in Oracle Lending* explains how UDFs can be defined and used. It details the procedure involved in defining UDFs and associating them with Products, so that the UDF is associated with contracts associated with all contracts involving the product. It also explains how to define UDFs and associating them with specific functions in Oracle Lending.
Chapter 3	*Function ID Glossary* has alphabetical listing of Function/Screen ID's used in the module with page references for quick navigation.

1.4 Glossary of Icons

This User Manual may refer to all or some of the following icons.

Icons	Function
☒	Exit
+	Add row
−	Delete row
🔍	Option List

1.5 Related Documents

For further information on procedures discussed in the manual, refer to the following manuals on:

- Core Entities
- Core Services
- Common Procedures
- Products

2. Creating Custom Fields in Oracle Lending

2.1 Introduction

While working with Oracle Lending, there are additional fields that you would like to use either for your convenience or to suit the requirements of your bank. Adding to its flexibility, Oracle Lending now provides you the option to add fields based on your specifications to meet your needs.

This chapter contains the following sections:

- Section 2.2, "User Defined Fields Maintenance"
- Section 2.3, "Scope of the Field"
- Section 2.4, "Numeric Field"
- Section 2.5, "Text Field"
- Section 2.6, "Date Field"
- Section 2.7, "Making a Field Applicable to a Product"
- Section 2.8, "Creating UDF and linking to a Function ID"
- Section 2.9, "Populating the Values of UDF"
- Section 2.10, "Uploading User Defined Fields"

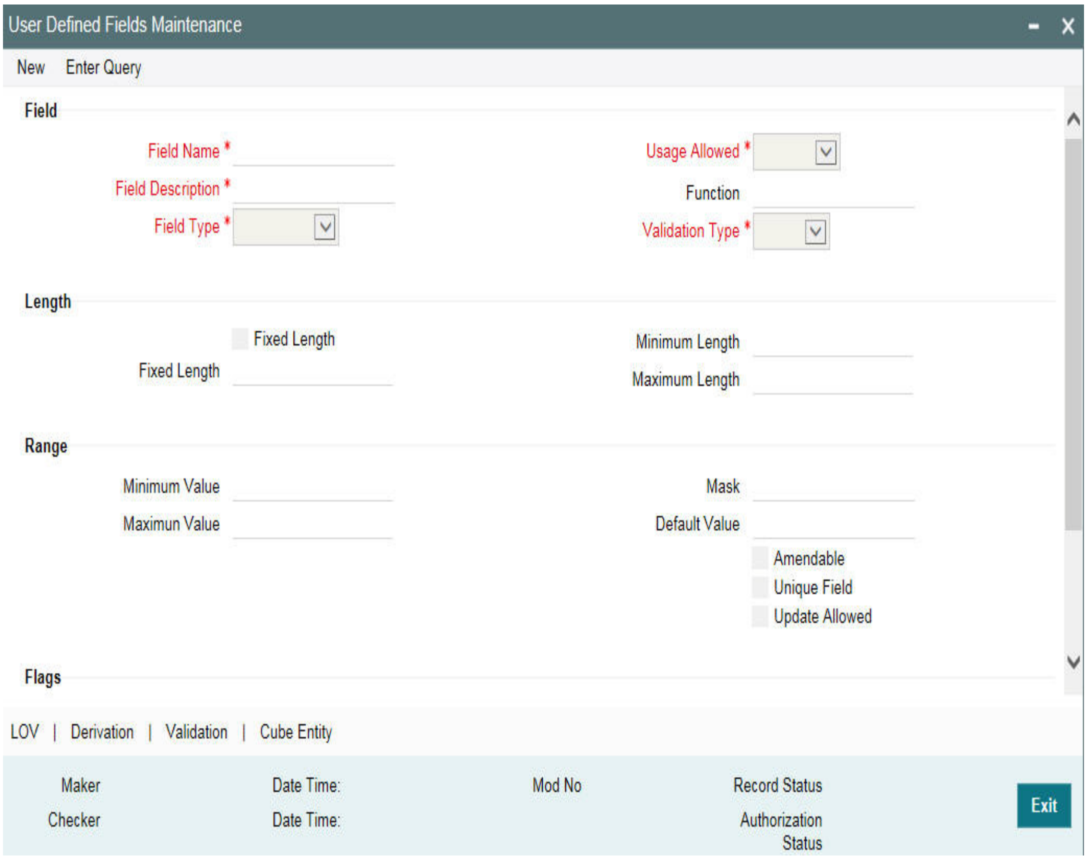
2.2 User Defined Fields Maintenance

This section contains the following topics:

- Section 2.2.1, "Invoking the User Defined Fields Maintenance Screen"
- Section 2.2.2, "Basic Details of User Defined Field Maintenance Screen"

2.2.1 Invoking the User Defined Fields Maintenance Screen

Based on your requirement and the nature of the field, you can specify default values and validations for the field. Oracle Lending validates all entries made to the field with the validations you define for a field. You can invoke the ‘User Defined Fields Maintenance’ screen by typing 'UDDUDFMT' in the field at the top right corner of the Application tool bar and clicking on the adjoining arrow button.



A field that you have created becomes operational in Oracle Lending only after it is authorized. A user bearing a different Login ID can authorize a field definition record that you have created.

2.2.2 Basic Details of User Defined Field Maintenance Screen

Field

Field Name and Description

You can identify a field that you create with a unique identifier, and a brief description.

Each field that you define in Oracle Lending should be assigned a unique code. You can briefly describe the field in Description field. The description is for your information only. Itis not be printed on any customer correspondence.

Field Type

The type of field that you can create in Oracle Lending can be of the following formats:

- Number - Choose this option to create a Numeric field
- Text - Choose this option to create a Text field
- Date - Choose this option to create a Date field
- Cube Entity - Should you need to reuse any of the existing fields of Oracle Lending, to enter additional details you can indicate the Field Type as Cube Entity. The Cube Entity can be any of the existing fields in Oracle Lending like customer, currency, account, and so on.

Marking a Field as Mandatory

You can make entry to a field mandatory. To do so, select the Mandatory option. You are forced to make an entry to the field. Leave it deselected to indicate that the field is not mandatory.

> **Note**
>
> When a UDF is created, you have to first map it to the corresponding function ID and then provide the validation rule for the UDF.

2.3 Scope of the Field

While defining a new user defined field, you need to specify whether the new field has to be used at the product or maintenance level. The scope or usage of the field that is being defined can be specified as ‘Usage Allowed’.

- If the new field that is being defined has to be used at the product level, select usage as ‘Product’
- If the new field is to be used in any of the maintenance screens (screens related to Core Maintenance like Currency Definition screen, Customer Accounts maintenance screen , and so on), select usage as ‘Function Id’.

> **Note**
>
> In Oracle Lending, every screen has a unique function Id. However to differentiate between product and maintenance levels, the scope of a user defined has been classified as Product and Function Id.

If the new field that is being defined is for a maintenance screen (Usage Allowed is ‘Function Id’), you can specify the function Id of the screen in which the new field has to be used. This has to be indicated in the ‘Function Id’ field. Click on the option list positioned next to this field. The function Ids of all the maintenance screens are displayed. Select the appropriate function Id.

For example, if you want to use the new field in the ‘Chart Of Accounts - Detailed’ screen, select the Function Id GLDCHACT, thus allowing the usage of the field in the Chart of accounts screen only.

> **Note**
>
> If you want the Field to be made available for all the Functions, you have to leave it blank.

2.4 Numeric Field

To define a numeric field, choose the number option at the Field Type field. You can set up validation rules for a numeric field. The validation types applicable to a numeric field are:

- Range
- Length
- LOV
- None

You can indicate your preference at the Validation type field of this screen. Choose *None* to indicate that no validation should be performed on entries made to this field.

Validation Type — Range

You can specify the range validation type only for Numeric fields. In this case the entry to the field should be within a permissible range.

On choosing this validation type, you should indicate either the maximum or minimum values or both values allowed for the field. Any valid entry to the field should be within the range that you specify.

Validation Type — Length

For a numeric field you can indicate that the entry should be of a certain length. You have the option to indicate,

- a fixed length
- the maximum and minimum length for the field

Depending on the option you select, indicate the fixed field length or indicate the maximum and minimum length for valid entries to the field.

Validation Type — LOV (List of Values)

Choose LOV to indicate that the entry to this field can be chosen from a predefined list. On choosing this option you can define the items that should be displayed on this list.

Click 'LOV' button and define the list of values and their description. The items that you define for the list are displayed whenever the field is used in Oracle Lending.

2.5 Text Field

This section contains the following topics:

- Section 2.5.1, "Specifying Text Fields"
- Section 2.5.2, "Validation Type — LOV (List of Values)"
- Section 2.5.3, "Derivation"
- Section 2.5.4, "Derivation Rule"

2.5.1 Specifying Text Fields

To define a text field, choose the text option at the Field Type field. A text field can contain alphabets of the English language or a combination of alphabets and numeric values.

You can specify validation rules for a text field. The validation types applicable to a text field include:

- Length
- Mask
- LOV
- None

You can indicate your preference at the Validation type field of this screen. Choose *None* to indicate that no validation should be performed on entries made to this field.

Validation Type — Length

For a text field you can indicate that a valid entry to the field should be of a certain length. You have the option to indicate,

- A fixed length
- The maximum and minimum length for the field

Depending on the option that you select, indicate the fixed field length or indicate the maximum and minimum length of entries made to the field.

Validation Type — Mask

To indicate a field as a masked field, choose Mask as the field type. This option allows you to define a broad field structure to which all entries to the field should conform.

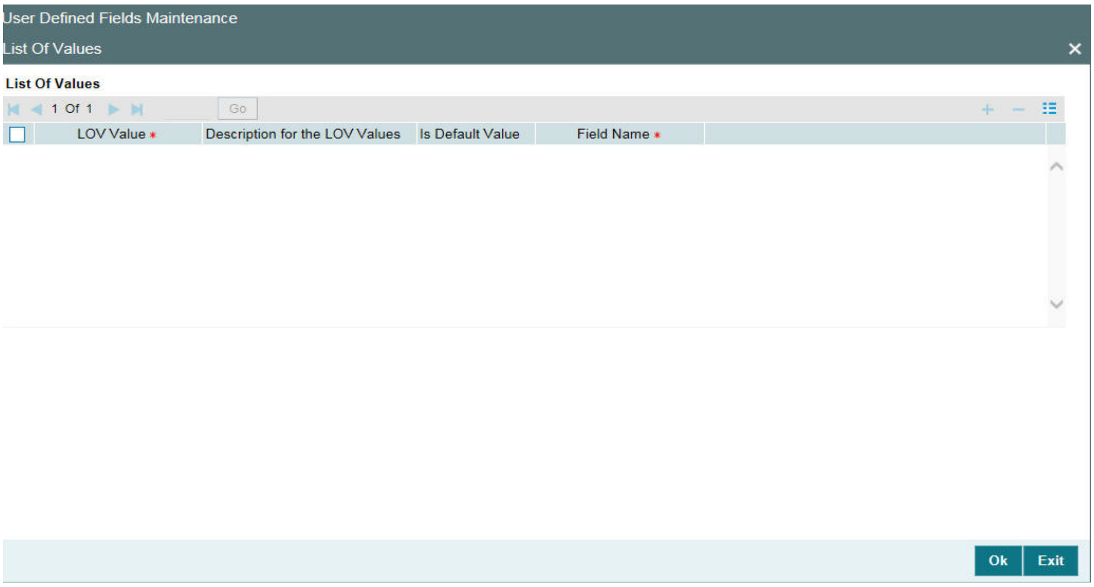
The mask structure can consist solely of ‘a’ or ‘n’ or a combination of these. An ‘a’ would indicate an alphabet of the English language and ‘n’ a numeric value.

All entries made to the fieldis validated with the format that you specify for the mask.

2.5.2 Validation Type — LOV (List of Values)

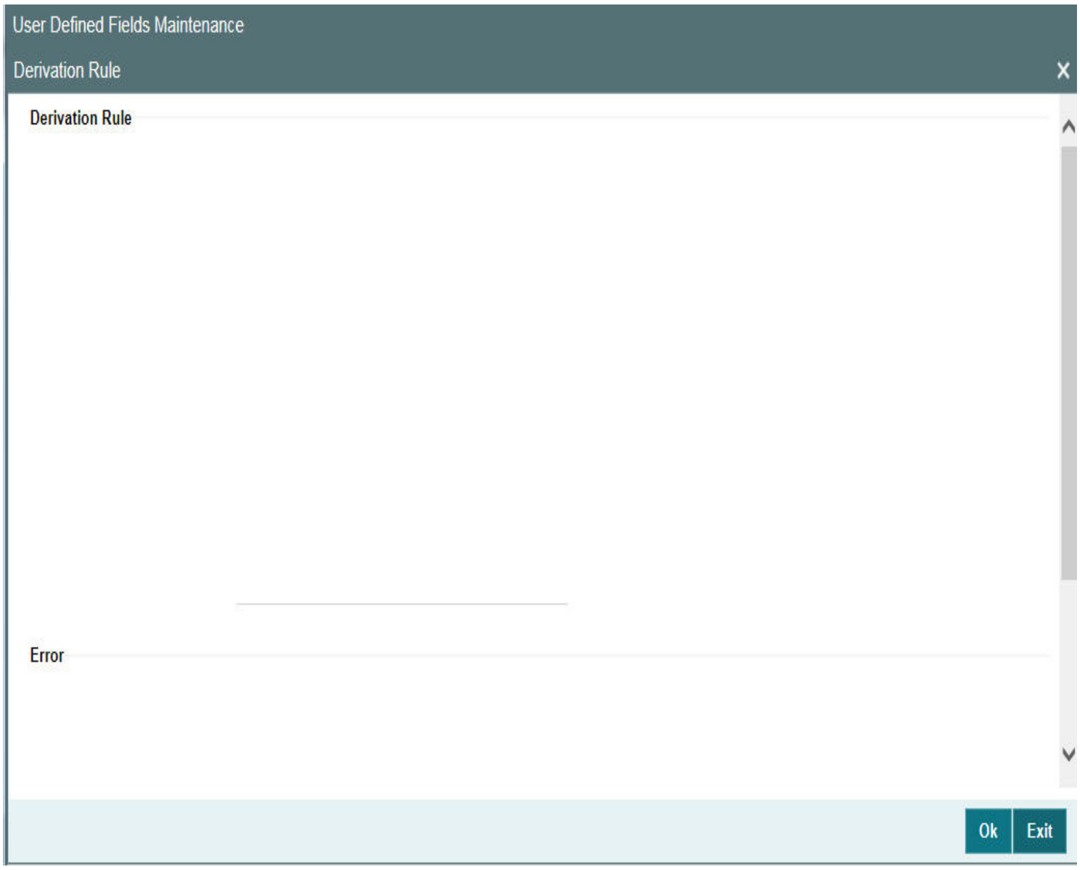
Choose LOV to indicate that the entry to this field can be chosen from a predefined list. On choosing this option you can define the items to be displayed on this list.

Click ‘LOV’ button and define the list of values and their description. The items that you define for the list are displayed whenever the field is used in Oracle Lending.



2.5.3 Derivation

This indicates the procedure for populating the values of a field. When you are processing a transaction that would use the user defined field, the value of the UDF can be populated in the ‘User Defined Field/Field Name to Value Definition’ screen that is invoked from the Contract Input screen or the Function Id screen. The value for the field can be populated based on certain conditions, which can be defined as statements of code by the user. Check against ‘Derivation Allowed’ to specify that the value of a field has to be populated based on certain conditions.



User Defined Fields Maintenance

Derivation Rule

2.5.4 Derivation Rule

Select the ‘Derivation Allowed’ option, if you want the values of a UDF to be populated based on certain conditions. According to the requirements of the bank, the implementer of Oracle Lending writes a PL/SQL code to populate the values of the user defined field. The values of the UDF are displayed in the ‘User Defined Field/Field Name to Value Definition’ screens of Contract Input or Function Id screen.

Click ‘Derivation’ button to write the PL/SQL code.

After writing the PL/SQL code, click ‘X’ button to execute the code. The derivation code is validated by the system. If any checks fail, you must alter the statement so that the validation can be made successfully. Click ‘E’ button to view the errors.

For example, the bank wants to have a new field to display the Euro equivalent of the contract amount in the ‘User Defined Field’ screen of Contract Input screen. To do this, you need to:

- Define a field of type ‘Number’,
- Specify ‘Usage Allowed’ as ‘Product’,
- Check against ‘Derivation Allowed’, and
- Write a code to display the Euro equivalent of the contract amount.

Subsequently, you can link the UDF to a product and process transactions under it. In the ‘User Defined Field’ screen of Contract Online screen, system executes the derivation rule to convert the transaction amount to Euro equivalent and displays the Euro equivalent of the transaction amount.

After writing the PL/SQL code, click ‘X’ button to execute the code. The derivation code is validated by the system. If any checks fail, you must alter the statement so that the validation can be made successfully. Click ‘E’ button to view the errors.

2.6 Date Field

This section contains the following topics:

- Section 2.6.1, "Validations that you can Specify for Date Fields"
- Section 2.6.2, "Marking the Field as Unique"
- Section 2.6.3, "Specifying the Default Value"
- Section 2.6.4, "Allowing the Modification of Values after Population"
- Section 2.6.5, "Factory Shipped Fields"
- Section 2.6.6, "Defining Additional Validation Rules"
- Section 2.6.7, "Validation Rule"
- Section 2.6.8, "Cube Entity"
- Section 2.6.9, "Enabling the Update Allowed Option for a UDF"

2.6.1 Validations that you can Specify for Date Fields

To define a date field, choose Date as the Field Type. For a date field, you can indicate validations like whether back and future dates can be entered into the field and the back or future period applicable to the field.

2.6.1.1 Indicating the Validation Type

You can specify the validation type for a date field. The validation types applicable to a date field include:

- LOV
- None

If you indicate *LOV* then an entry to the field can be made only from the predefined list that you maintain for the field. Choose *None* to indicate that no validation should be made for the field.

Back/Future Date Allowed

For a date field you can indicate whether back or future dates can be entered. Select the relevant options to indicate your preference.

If you choose the back or future date options, you should also indicate a future or back period permissible for the field. The back or future period should be represented as a number.

For example, if you indicate ‘3’ as the back date period, the field accept dates upto three days before the current system date as a valid entry.

Note

If you do not choose any of these options, the field only accept the current system date as a valid entry.

2.6.2 Marking the Field as Unique

Select ‘Unique’ if the field that is being defined has to be unique. Consequently,

- If the usage allowed for the field is ‘Product’, the new field is unique across modules
- If the usage allowed is ‘Function Id’, it is unique for the particular function id that is., if you have specified a function ID, the new field can be used only in the specified Function ID screen).

2.6.3 Specifying the Default Value

You can specify a default value for a user-defined field. The option list positioned next to this field is enabled only if the field is a cube entity. Otherwise, the option list is disabled.

For example, assume that your bank wants to capture an additional currency field for processing certain transactions and the default value for this additional field has to be USD. Therefore, your selections are as follows:

- Field Type –Cube Entity
- Cube Entity Type- Currency

The option list in the ‘Default Value’ field displays all the currencies maintained at your bank. Select USD. Consequently, in the UDF screen of the Contract Input or Function Id screen, USD is displayed as the default currency.

Note

If you have specified a default value for a field and also the derivation rule, the value obtained from the derivation rule takes precedence over the default value at the Contract/ Function Id level.

2.6.4 Allowing the Modification of Values after Population

You can modify the value of a field after it is populated in the ‘User Defined Field’/’Field Name to Value Definition’ screen of Contract Input or Function Id screen. To allow amendments to the values after they are populated, select the field ‘Amendable’ at the time of defining a new field.

Note

You are not allowed to change the values of those fields for which you have disallowed the amendment option (if the option ‘Amendable is unchecked at the time of defining an UDF) and derivation has been allowed (Derivation Allowed option has been checked).

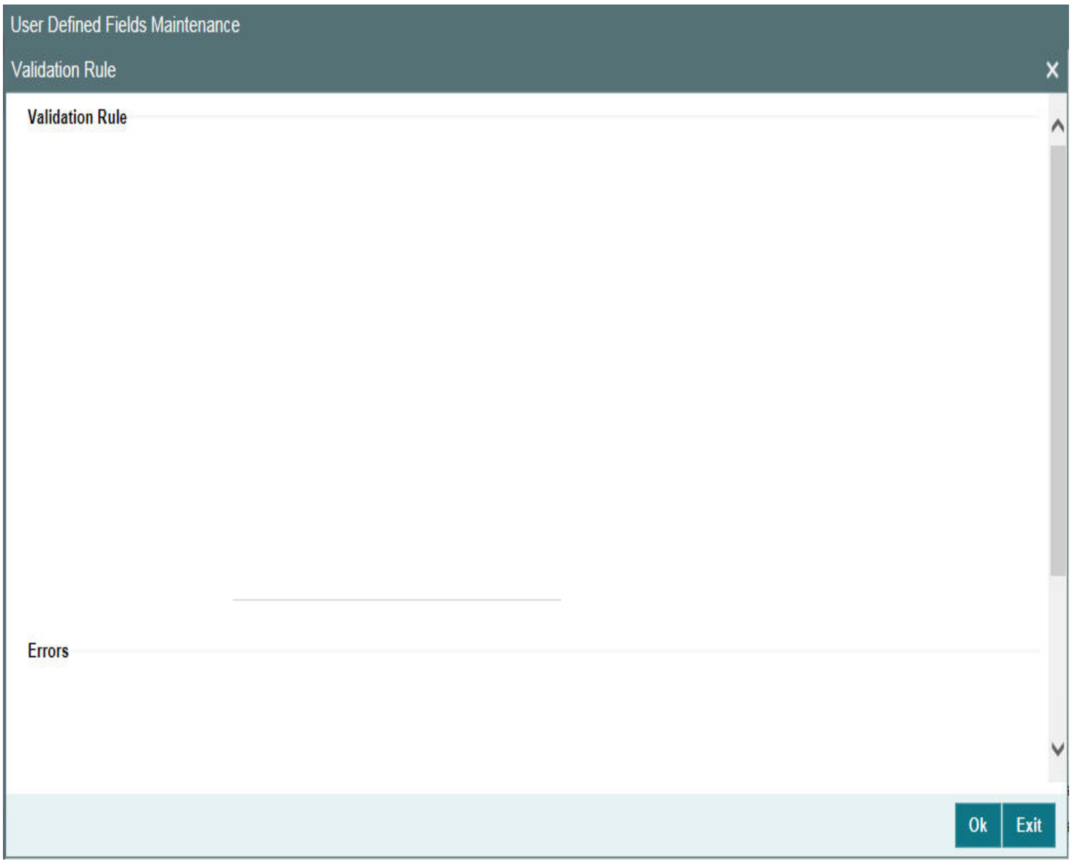
2.6.5 Factory Shipped Fields

Apart from the fields that you have created, there are a set of fields that are sent by default as part of Oracle Lending. These fields are referred to as factory shipped fields. You cannot define or change validations specified for factory shipped fields.

2.6.6 Defining Additional Validation Rules

Apart from specifying the validation type for a field, you can specify additional validation rules to meet the requirements of the bank.

To indicate that additional validations are required for a particular user defined field, select ‘Validation Allowed’. This allows you to write a code to validate the value in the new field. Click ‘Validation’ button to write the validation code.



2.6.7 Validation Rule

This is the PL/SQL validation code based on which the system check the value of the new field and validates at the time of transaction processing. Entry to this field is mandatory if you have selected ‘Validation Allowed’ option.

For example, at the time of processing a contract for a customer, the bank wants to add a new field to enter the first nominee that the customer has specified in his account. The names of the nominees of a customer are maintained in the ‘Customer Accounts Maintenance’ screen. The bank wants the system to check that the name of the nominee that is entered in the new field and the name of the first nominee that is maintained in the Customer Accounts Maintenance screen are same.

To do this, you need to:

- Create a new user defined field of type ‘Text’,
- Specify ‘Usage Allowed’ as ‘Product’,
- Check against ‘Validation Allowed’, and
- Write a code to validate the value entered in the new field. In the validation rule, you write a code to check that the value of the first nominee specified in the Customer Accounts Maintenance screen for the customer for whom you are processing a contract and the value entered at the time of processing a contract are same.

Syntax to be used in Derivation and Validation Rules
(@FIELD_VAL)

(@FIELD_VAL) holds the Value UDF. This is s mandatory for Derivation Logic and it can be used in Validation logic to get the value of the field.

For example,

(@FIELD_VAL): = ‘USD’;

Select ccy_code into (@FIELD_VAL) from

CYTMS_CCY_DEFN where country = ‘USA’;

(@RECORD_KEY)

(@RECORD_KEY) behaves differently for UDFs’ linked to Product and different for UDFs’ linked to Function ID.

(@RECORD_KEY) When UDF is linked to FUNCTION_ID

To use (@Record_key) in UDFs’ where Usage allowed is ‘Function_id’ you have to maintain Function Key Mapping, which will be used to determine the record key before you define the UDF.

2.6.8 Cube Entity

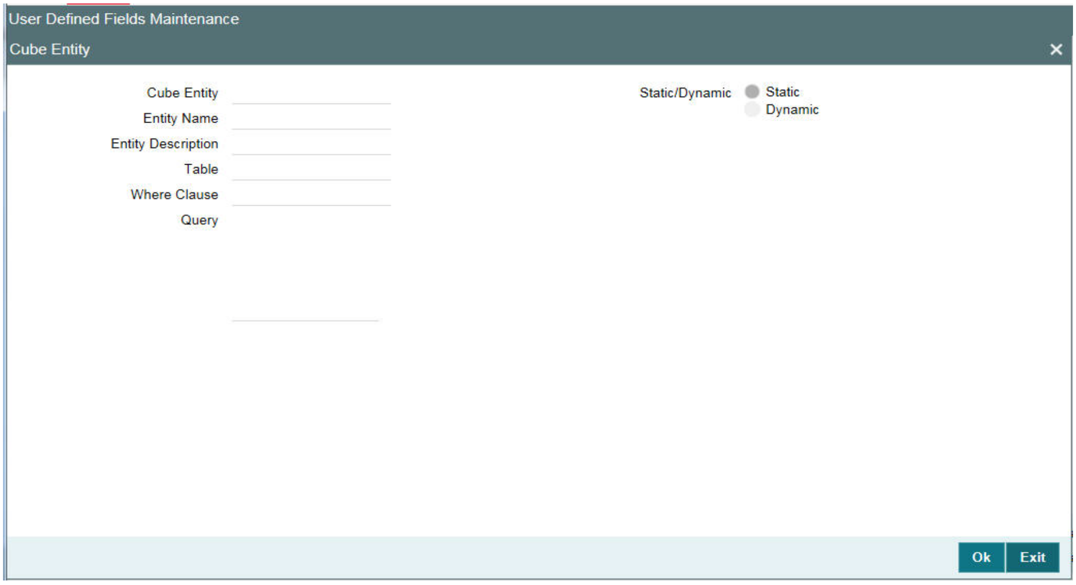
If the field that you are creating is of the type **Cube Entity**, indicate the entity that is applicable to the field. You can select a Cube Entity from the option list positioned next to this field. This contains the following Cube Entities:

- Currency
- GL
- Branch
- Customer
- Account
- Contract Ref No
- User Ref No
- Liability ID

For instance, you need to capture details of an additional customer for a transaction. You can choose the Cube Entity field type and select the Customer field. The list of customers maintained for your bank are made available at the field as in any other Customer field in Oracle Lending.

Method for Populating the Cube Entity Values

If the field type is ‘Cube Entity’, the values of a field can be selected from Oracle Lending tables. If you select the field type as ‘Cube Entity’, the ‘Cube Entity’ screen is displayed.



In the ‘Cube Entity’ screen, you need to specify the method by which the values have to be populated. The options available are:

- Static – If you choose this option, system retrieves the values from a Oracle Lending table. After specifying the cube entity, specify the table name, column name and where clause based on which the data is retrieved from the table. If you have defined a cube entity once, you can reuse it. The adjoining option list displays all the cube entities, which you had used earlier displays. Select the appropriate cube entity. On selection of the reused cube entity, system automatically displays the table name, column name and where clause (if specified for the selected cube entity). You can however modify the condition in the where clause but not the table name and column name.
- Dynamic - If you choose this option, the values on the UDF are populated based on the query written by the user

Note

At the time of installation, the implementer writes the PL/SQL code for the derivation rule, validation rule and also the query for fetching a cube entity according to the requirements of the bank.

Table

Depending on your selection in the ‘Cube Entity’ field, system displays the table name in which the selected cube entity is located. The value of the column (which is specified in the previous field) from this table is picked up to populate the values of the new field.

Description

After you enter the cube entity, you may enter a brief description of the cube entity. This description is used for information retrieval.

Where Clause

Enter the condition based on which the values from the specified column and table, the values of the new field should be picked up. Enter the condition in this field if you have indicated ‘Static’ method for populating the values of the cube entity.

Query

This is the code, based on which the values of the new field is picked up.

The query has to be written if the option ‘Dynamic’ is selected.

2.6.9 Enabling the Update Allowed Option for a UDF

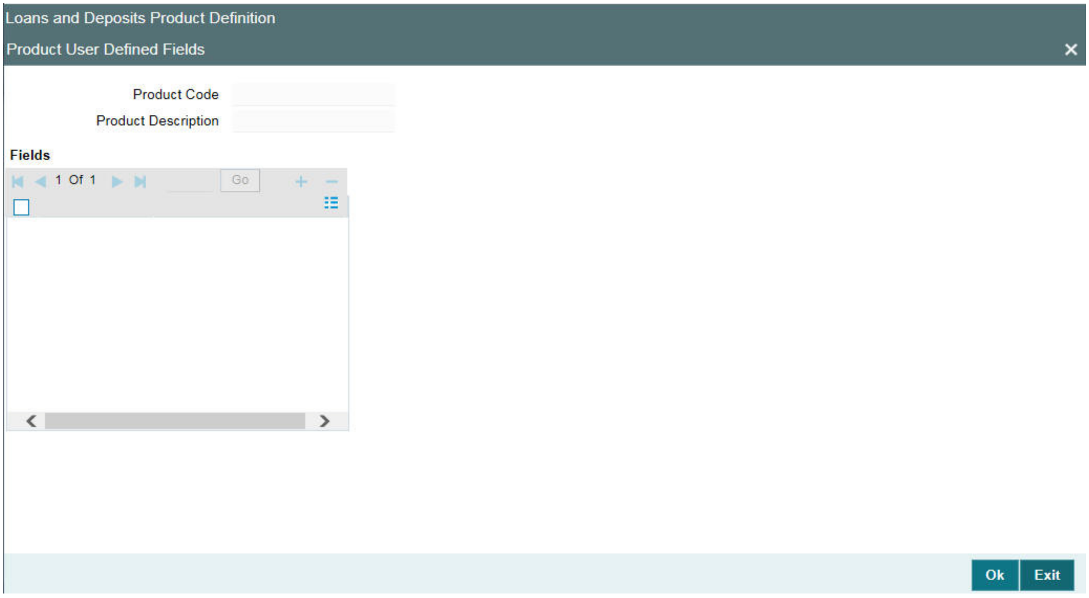
The Update Allowed option is used during Event Processing. You can assign values to a UDF when a UDE is processed. During event processing, the system assigns certain UDF parameters based on the Execution Query you have maintained in the Event Processing screen. You are allowed to assign UDFs only for those UDFs for which the Update Allowed option has been enabled.

2.7 Making a Field Applicable to a Product

The fields that you define in the User Defined Fields screen can be made applicable to the products (and thereby to the contracts) that you create for the front-end modules of Oracle Lending depending on your selection in the ‘Usage Allowed’ field.

For example, suppose you have defined a user-defined field DATE1 with ‘Usage Allowed’ as Product, the UDF is displayed for association only in the Product Definition screen of a module.

At the time of creating a product click ‘Fields’ button from the ‘Product Definition’ screen. From this list of all the fields that you created, you can choose specific fields and make them applicable to the product.


Loans and Deposits Product Definition
Product User Defined Fields
×
Product Code
Product Description
Fields
1 Of 1
Go
+
−
□
Ok
Exit

In this screen, you can link the user-defined fields to the product that you are defining. Oracle Lending allows you to link up to two hundred fields to a product.

To link a user-defined field, click add icon. A list of all the user-defined fields for which the Usage Allowed is ‘Product’ is displayed. Select the fields that have to be included in the Contract Input screen for the contracts processed under the product that is being defined.

When a contract is processed under that product, the fields associated with the product are available in the Contract Input screen.

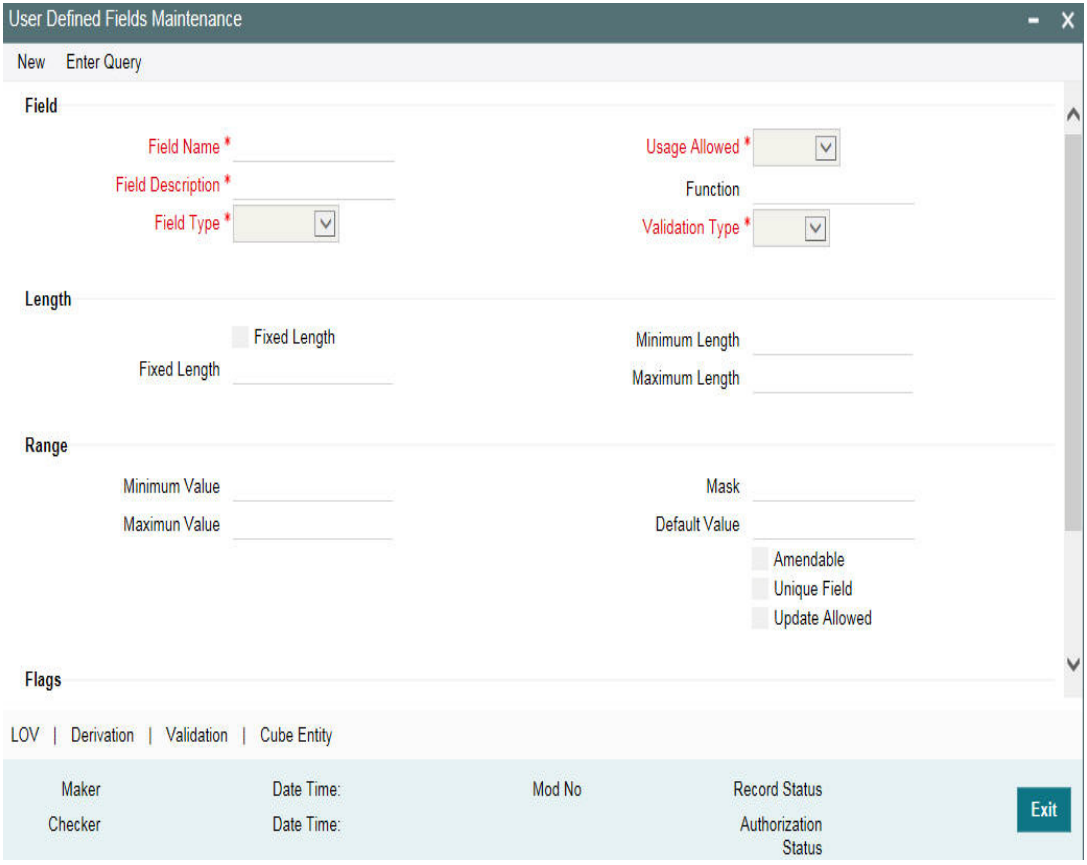
2.8 Creating UDF and linking to a Function ID

This section contains the following topics:

- Section 2.8.1, "Creating UDF"
- Section 2.8.2, "Linking UDF to Function ID"

2.8.1 Creating UDF

You can create UDFs using the ‘User Defined Fields Maintenance’ screen for a particular function id. You can invoke this screen by typing ‘UDDUDFMT’ in the field at the top right corner of the Application tool bar and clicking the adjoining arrow button.



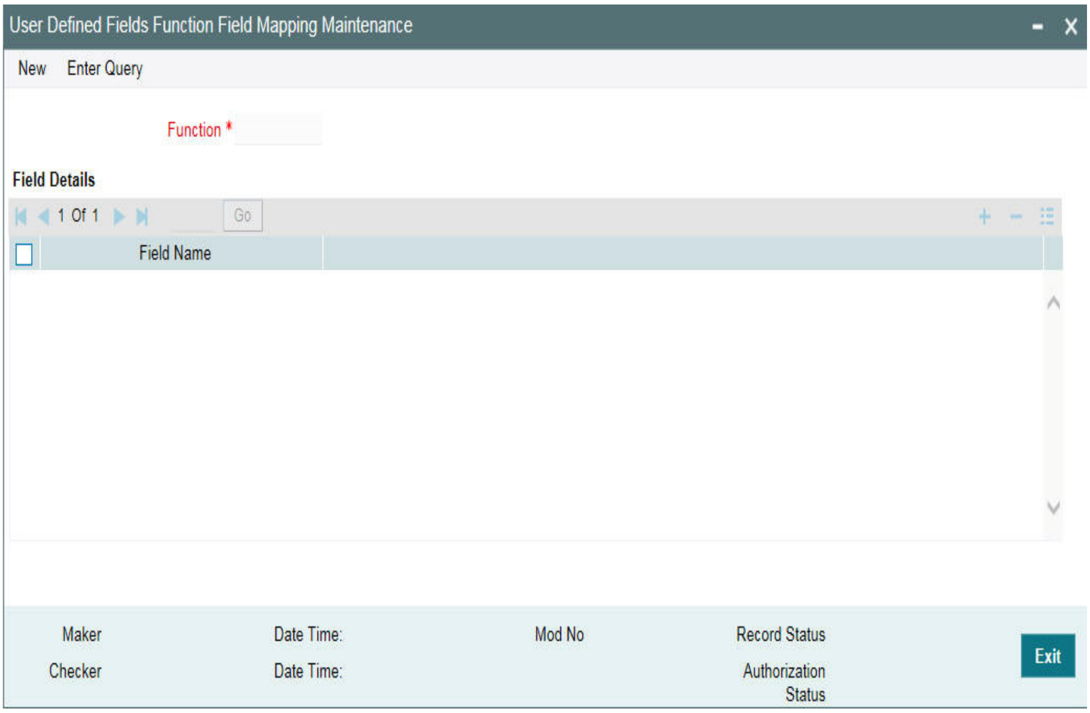
If you select ‘Usage Allowed’ as ‘Product’, then you need not provide function ID. You can fetch these fields in the ‘Product Definition’ screen.

If you select ‘Usage Allowed’ as 'Function', then you need to provide the function id in the ‘Function’ field.

After saving and authorizing the record in the ‘User Defined Fields Maintenance’ screen, go to ‘User Defined Fields Function Field Key Mapping Maintenance’ screen.

2.8.2 Linking UDF to Function ID

You can invoke the ‘User Defined Fields Function Field Key Mapping Maintenance’ screen by typing ‘UDDFFLMT’ in the field at the top right corner of the Application tool bar and clicking on the adjoining arrow button.



In this screen, you must identify the function ID to which you need to link user-defined fields.

Select the Function Id from the adjoining option list. For instance, if you want to include the user defined fields in the 'Chart Of Accounts- Detailed' screen, select the function Id GLDCHACT. After selecting the Function Id, click 'Add' icon to select the fields that have to be included in the selected Function Id screen. A list of all function Id 's for which the Usage Allowed is specified as 'Function Id' will be displayed. Select the appropriate fields that have to be included from the list. Oracle Lending allows you to link up to two hundred fields to a Function ID.

After you save and authorize the record, the user defined fields will be included in the specified Function Id screen.

2.9 Populating the Values of UDF

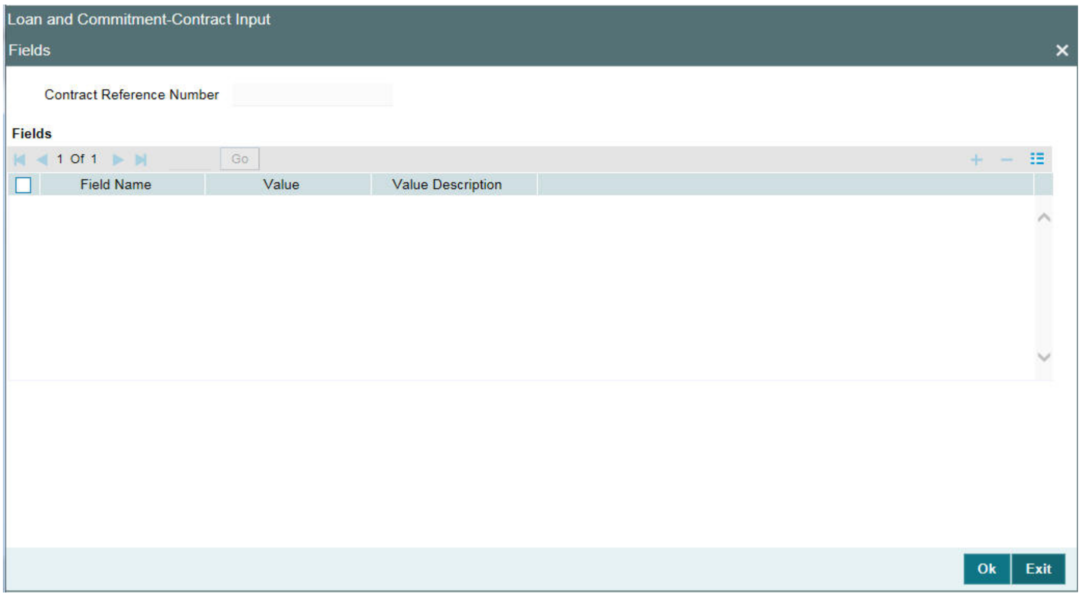
This section contains the following topic:

- Section 2.9.1, "Population of UDF Values at the Contract Level"

2.9.1 Population of UDF Values at the Contract Level

The defaulted or derived values of the UDF linked at the product level are populated in the Contract Input screen at the time of processing contracts under a particular product. Similarly, the values of the UDF's linked to a function Id are populated in the function Id screen.

The UDF's linked to a particular product are available when a contract is processed under that product. In the Contract Input screen, click 'Fields' button to invoke the 'User Defined Fields' screen.



In this screen, all the user defined fields associated with the product under which you are processing the contract are displayed.

According to your specifications at the time of defining the user defined fields, system may derive the values of the UDF from:

- Derivation Logic: When you click 'Fields' button, system derives and displays the values of the fields for which derivation rules are written. The value description is also displayed alongside.
- Static List of Values: If the validation type of the UDF that is associated with the contract is 'List of Values' (LoV), systemdisplays the static values, along with the value description. If you have indicated a default value for the LoV, the default value and its description is displayed. When a user selects the value from the LoV, the system displays the description on the screen.
- Method of derivation for cube entity fields: If a UDF is of type Cube Entity, system populates the values in a LoV. The values will be derived from the specified Oracle Lending table (if the cube entity type is 'Static') or from the query written by the user (if the cube entity type is 'Dynamic')

You are not allowed to change the values of the fields for which you have not allowed amendment (if the option 'Amendable is unchecked at the time of defining an UDF) and derivation is allowed (Option 'Derivation Allowed' is checked).

Enter the values for the fields, for which the values are not derived by the system.

You can change the value of a UDF after the system has derived the value from the derivation logic. But it might so happen that the derivation logic of another UDF might use the value of the UDF that you have changed. Consequently, if you change the value of the UDF whose value is used in another UDF, the value of that UDF will also change.

For example, let us assume that you have defined a user defined field UDF1 to display the Euro equivalent of the contract currency.

The derivation logic of another user defined field (UDF2) utilizes the values of UDF1 to arrive at the value of UDF2. Assume that in the derivation logic written for UDF2, you have specified that the value of UDF2= 2 UDF1.

At the contract level, system converts the contract currency into Euro equivalent and displays the value of UDF1, as 220.00. According to this value of UDF2 will be 444.00. However, if you change the value of UDF1 to 250.00, it effects on the value of UDF2 also (it becomes 500.00).

If the change in the UDF value has effected any other UDF, system will display an override informing you about the UDF whose value will be changed. Select ‘Ok’ button if you want the system to re-calculate the value of the UDF based on the modified UDF value.

2.10 Uploading User Defined Fields

This section contains the following topics:

- Section 2.10.1, "Contract Upload"
- Section 2.10.2, "Function Upload"

2.10.1 Contract Upload

Your bank may use Oracle Lending to process certain transactions (say OL transactions) and use another application to process other transactions (say MM transactions).

At some point of time, your bank may want to use the values from the external system in Oracle Lending. For this purpose, Oracle Lending stores the values of the various fields from the external system (an application other than Oracle Lending) in a database table. If your bank wants to use the value of the fields from the external system in Oracle Lending, the values have to be brought into Oracle Lending database. This is done through a process called ‘Upload’.

Let us assume that your bank is processing OL transactions in Oracle Lending and MM transactions in an external system. For processing certain OL transactions, your bank wants to use the values from certain MM transactions. Therefore, you have to put the value of the contract reference number along with the field name and value of the MM transactions, which you want to use in Oracle Lending into a database Upload table with the status marked as ‘U’ (indicating that it has to be taken up for processing). This Upload table contains the values posted from the external system. Subsequently, you have to bring the values of these fields into Oracle Lending. This is done by running the contract upload program.

When you run this program, Oracle Lending selects the fields for which the status is ‘P’. It uploads the values from the external system into Oracle Lending database.

2.10.2 Function Upload

The values of the fields from external system used in the maintenance screens should be uploaded into Oracle Lending database.

3. Function ID Glossary