Oracle® Banking APIs Cloud Service

Auto Categorization of Spend Transactions

Release 25.1.0.0.0
G28949-01
April 2025

Oracle Banking APIs Cloud Service Auto Categorization of Spend Transactions, Release 25.1.0.0.0

G28949-01

Contents

Preface

Purpose

This guide is designed to help acquaint you with the Oracle Banking Digital Experience application. This guide provides answers to specific features and procedures that the user need to be aware of the module to function successfully.

Audience

This document is intended for the following audience:

- Customers
- Partners

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.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve.

Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1 Acronyms and Abbreviations

Abbreviation	Description
OBAPI	Oracle Banking APIs

1

Auto Categorization Of Spend Transactions

Scenario:

If a Spend transaction description contains keyword "Big Bazaar" then that transaction should get auto-categorized as follows:

Category → Shopping

Subcategory → Groceries

Maintenance steps:

1. Create a Category and a Subcategory as required after login as an Admin user.
2. Note the `CATEGORY_ID` column value of the newly created Category & Subcategory from the table `DIGX_SP_SPEND_CATEGORY` (Eg: 10 and 101 respectively).
3. Insert a row in table `DIGX_RL_SPEND` as below (Note the ID column value which is '10001' in this example).
   Sample script:

```
insert into DIGX_RL_SPEND
        (ID, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE,
        OBJECT_STATUS,   OBJECT_VERSION_NUMBER, NAME, DESCRIPTION,
DETERMINANT_VALUE)
        values ('10001', '', sysdate , '', sysdate , '', 1,'SpendRule',
'SpendDTODesc', 'OBDX_BU');
```

4. Insert a row in table `DIGX_RL_CONDITION` as below:
   a. ID as any unique value
   b. RULE_ID as provided in the `DIGX_RL_SPEND` table above
   c. EXPRESSION_OBJECT value should contain the keyword as follows:

```
utl_raw.cast_to_raw('{ "@class" : "

com.ofss.digx.app.spendanalysis.dto.rule.SpendCategorizationExpressionFa
ctValueDTO",
        "keyword" : "Big Bazaar" }')
```

   d. `OUTCOME_OBJECT` value should contain the corresponding Category & Subcategory ID as follows (Subcategory is NOT mandatory):

```
utl_raw.cast_to_raw('{ "@class" : "

com.ofss.digx.app.spendanalysis.dto.rule.SpendCategorizationOutcomeFactV
alueDTO",
      "category" : "10", "subCategory" : "101" }')
```

e. OPERATOR value should be 'CONTAINS' (No other operators are supported out of the box
Sample script:

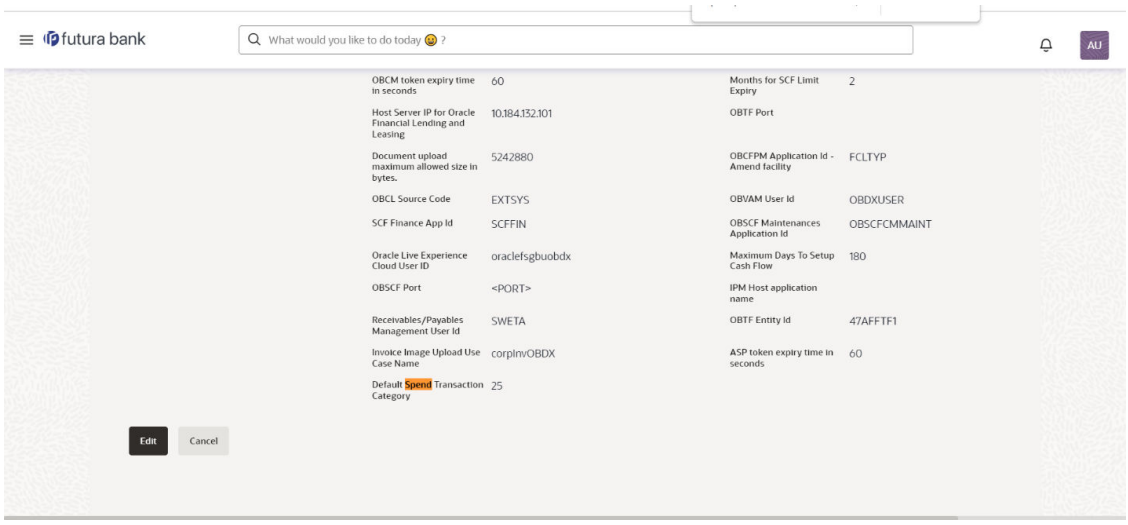
```
insert into
        DIGX_RL_CONDITION (ID, RULE_ID, OPERATOR, OUTCOME_OBJECT,
EXPRESSION_OBJECT, CREATED_BY,
        CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE,
OBJECT_STATUS, OBJECT_VERSION_NUMBER,
        DETERMINANT_VALUE)values ('103', '10001',
        'CONTAINS', utl_raw.cast_to_raw('{"@class" :

"com.ofss.digx.app.spendanalysis.dto.rule.SpendCategorizationOutcomeFact
ValueDTO","category" :
        "10","subCategory" :
        "101"}'),
        utl_raw.cast_to_raw('{"@class" :

"com.ofss.digx.app.spendanalysis.dto.rule.SpendCategorizationExpressionF
actValueDTO","keyword" : "Big
        Bazaar"}'), '', null, '',
        null, '', 1, 'OBDX_BU');
```

5. Insert such rows in table `DIGX_RL_CONDITION` for each keyword & Category-Subcategory combination as required by the Bank.
6. Verify the following properties (select * from `DIGX_FW_CONFIG_ALL_B` where category_id = 'ruleconfig';):
   a. `IS_RULE_EVALUATION_REQUIRED` → Y
   b. `SPEND_EVALUATOR` → com.ofss.digx.app.spendanalysis.ruleengine.evaluator.SpendRuleEvaluator (out of the box Spend rule evaluator)
   c. `SPEND_RULE_NAME` → SpendRule (As per the value of NAME column in DIGX_RL_SPEND table)
   d. SPEND_DEFAULT_CATEGORY_ID → This property is used for maintaining default category, if no keyword is match from all maintained rules.

Below attached screenshot entry is required for setting default category. We need to update the entry from day one configuration screen. We have to maintain the Category ID in this configuration value.

> **Note:**
>
> If below entry is not maintain and no rule is matched, then no category is set against the spend transaction.

Figure 1-1 "System configuration" screen with required entry for setting default category



2

Categorization Of Spend Transactions Using ML Engines

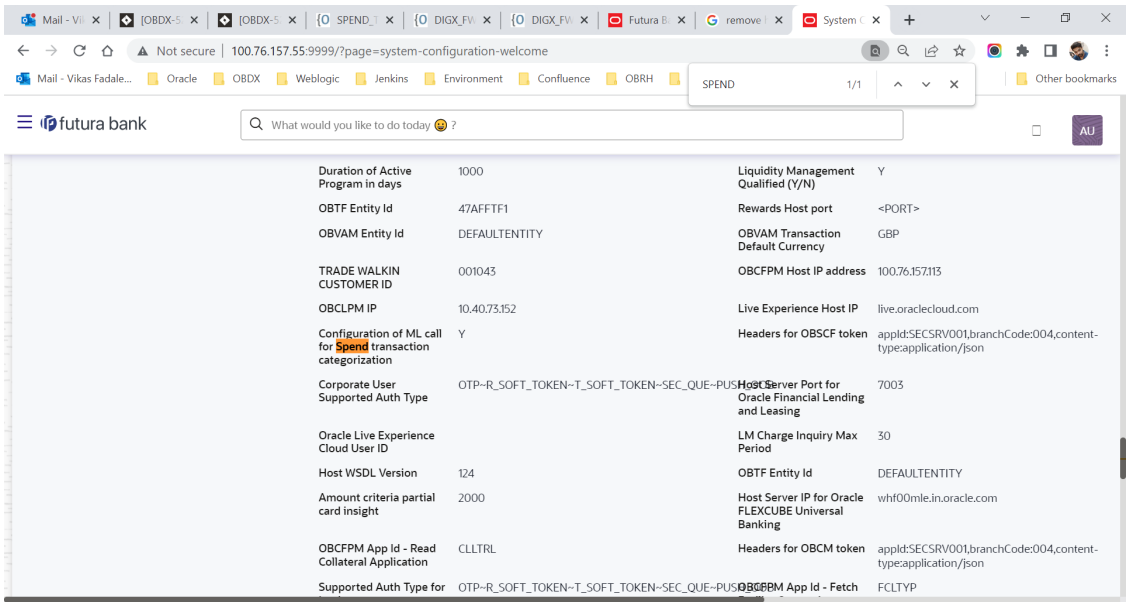
ML Engine will do the auto categorization on user spend expenditure. It will auto map the category base on transaction description. If in case category does not exist in the system, then first it will create it and then map it to the transaction.

To activate the ML engine call, we have to set “**Configuration of ML call for Spend transaction categorization**” flag as **Y**. If in case, we don’t set this value then by default ML engine service is in inactive state.

We can be able to update the flag with the help of **“System configuration”** screen with Admin Type Role.

Below entry are required to activate ML engine interface call.

Figure 2-1 System configuration screen with Admin Type Role




```
Insert into DIGX_FW_CONFIG_ALL_O

(PROP_ID,PREFERENCE_NAME,PROP_VALUE,DETERMINANT_VALUE,CREATED_BY,CREATION_DATE
,LAST_UPDATED_BY,LAST_UPDATED_DATE)
        values

('<<ENTITY_ID>>.com.ofss.digx.extxface.spendanalysis.adapter.transaction.ISpen
dUnCategorizedTransactionAdapter.getSpendTransactionCategory','ExtxfaceAdapter
Preference','com.ofss.digx.extxface.spendanalysis.impl.SpendDDATransactionAdap
ter','01','ofssuser',to_timestamp(sysdate,'DD-MON-RR
        HH.MI.SSXFF AM'),'ofssuser',to_timestamp(sysdate,'DD-MON-RR
```

```
HH.MI.SSXFF
        AM'));

Insert into DIGX_FW_CONFIG_ALL_O

(PROP_ID,PREFERENCE_NAME,PROP_VALUE,DETERMINANT_VALUE,CREATED_BY,CREATION_DATE
,LAST_UPDATED_BY,LAST_UPDATED_DATE)
        values

('SPEND_DDA_TRANSACTION_CATEGORIZATION','ExtXfaceConfig','com.ofss.digx.extxfa
ce.impl.endpoint.OBRHEndPoint','<<ENTITY_ID>>','ofssuser',to_timestamp(sysdate
,'DD-MON-RR
        HH.MI.SSXFF AM'),'ofssuser',to_timestamp(sysdate,'DD-MON-RR
HH.MI.SSXFF
        AM'));
```

Index

A

C