

Mid-Office Product Setup and Configuration Guide
Oracle Banking Digital Experience Cloud Service
Release 22.2.1.0.0

Part No. F76129-01

May 2023

ORACLE®

Mid-Office Product Setup and Configuration Guide

May 2023

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/

Copyright © 2006, 2023, Oracle and/or its affiliates. All rights reserved.

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

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 failsafe, 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.

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.

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

Table of Contents

| | |
|--|-------------|
| 1. Preface | 1-1 |
| 1.1 Intended Audience | 1-1 |
| 1.2 Documentation Accessibility | 1-1 |
| 1.3 Access to Oracle Support | 1-1 |
| 1.4 Structure | 1-1 |
| 2. Introduction | 2-2 |
| 3. Trade Finance | 3-1 |
| 3.1 Oracle Banking Trade Finance (OBTF) | 3-1 |
| 3.2 Oracle Banking Trade Finance Process Management (OBTFPM) | 3-7 |
| 4. Corporate Lending | 4-1 |
| 4.1 Day One Executions | 4-1 |
| 5. Supply Chain Finance | 5-1 |
| 5.1 Non Customer Onboarding using chaining | 5-1 |
| 5.2 OBRH Integration | 5-3 |
| 6. Receivables/Payables Management | 6-1 |
| 6.1 OBRH Integration | 6-1 |
| 7. Virtual Account Management | 7-1 |
| 7.1 OBRH Integration | 7-1 |
| 7.2 Verify System Configurations | 7-4 |
| 7.3 Enumerations | 7-5 |
| 7.4 Adapter Properties | 7-5 |
| 8. Cash Management System | 8-1 |
| 8.1 OBRH Integration | 8-1 |
| 9. Credit Facility Management | 9-1 |
| 9.1 Core as Third Party | 9-1 |
| 9.2 OBRH Integration | 9-1 |
| 10. Liquidity Management | 10-1 |
| 10.1 OBRH Integration | 10-1 |
| 10.2 Verify System Configurations | 10-2 |
| 10.3 Enumerations | 10-3 |
| 10.4 Adapter Properties | 10-3 |

| | | |
|------------|---|-------------|
| 10.5 | Simulation IC Group maintenance..... | 10-3 |
| 11. | User Credential Configuration | 11-1 |
| 12. | OBRH Integration Configuration | 12-1 |

1.1 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

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 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

1.4 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describes following details:

- Introduction
- Preferences & Database
- Configuration / Installation.

2. Introduction

This document is intended for setting up OBDX 21.1.1.0.0 with different Mid-Office Products.

[Home](#)

3. Trade Finance

3.1 Oracle Banking Trade Finance (OBTF)

All host calls are OBRH enabled.

3.1.1 OBRH Mandatory Executions

Execute the following script at OBDX digx database and restart the managed server.

```
../installables/db/OBTF/DIGX_FW_CONFIG_ALL_O.sql
```

In case of patch set installation pls execute following script

```
../patch_incrementals/modules/OBTF/DIGX_FW_CONFIG_ALL_O.sql
```

Note: '%ENTITY_ID%' should be replaced with entity identifier (For example 'OBDX_BU')

3.1.2 OBRH Configurations

Following day 1 system configuration should be available at entity level to connect OBDX with OBRH system.

```
select * from digx_fw_config_var_b where prop_id in ('OBRH_EXPIRYTIME','OBRH_HOST_IP','OBRH_HOST_PORT');
```

Please find below the details of OBDX Trade Finance requests and their corresponding services which are available inside OBDX consumer of OBRH application.

| Trade Finance Request | Consumer Service |
|--|--|
| Trade applications list | TRADE_APPLICATIONS_LIST |
| Trade clarification list | TRADE_APPLICATION_CLARIFICATION_LIST |
| Trade application clarification update | TRADE_APPLICATION_CLARIFICATION_UPDATE |
| Download the document by given document ref id | TRADE_APPLICATION_DOCUMENT_DOWNLOAD |
| Link already uploaded documents to a contract | TRADE_APPLICATION_DOCUMENT_LINKAGE |
| Trade application document list | TRADE_APPLICATION_DOCUMENT_LIST |
| Trade application document upload | TRADE_APPLICATION_DOCUMENT_UPLOAD |
| Trade application list | TRADE_APPLICATION_LIST |

| Trade Finance Request | Consumer Service |
|--|----------------------------------|
| Trade bank code list | TRADE_BANK_CODE_LIST |
| Trade bank details | TRADE_BANK_DETAILS |
| Trade base date list | TRADE_BASE_DATE_LIST |
| BC product currencies | TRADE_BC_PRODUCT_CURRENCIES |
| BC product list | TRADE_BC_PRODUCT_LIST |
| BC product read details | TRADE_BC_PRODUCT_READ_DETAILS |
| BG amend charges simulation | TRADE_BG_AMD_CHARGES_SIMULATION |
| BG amend create | TRADE_BG_AMEND_CREATE |
| BG amend list | TRADE_BG_AMEND_LIST |
| BG pending amendment listing | TRADE_BG_AMEND_PENDING_LIST |
| BG amend read | TRADE_BG_AMEND_READ |
| BG availment listing | TRADE_BG_AVAILMENT_LIST |
| BG bank customer read | TRADE_BG_BANK_CUSTOMER_READ |
| BG charges simulation | TRADE_BG_CHARGES_SIMULATION |
| BG claim create | TRADE_BG_CLAIM_CREATE |
| BG create | TRADE_BG_CREATE |
| BG swift and advice message read details | TRADE_BG_LEVEL_ADVICE_SWIFT_READ |
| BG charges listing | TRADE_BG_LEVEL_CHARGES |
| BG listing | TRADE_BG_LIST |
| BG pending amendment acceptance | TRADE_BG_PENDING_AMEND_ACCEPT |
| BG pending amendment check | TRADE_BG_PENDING_AMEND_CHECK |
| BG pending amendment rejection | TRADE_BG_PENDING_AMEND_REJECT |
| BG product currency listing | TRADE_BG_PRODUCT_CURRENCY_LIST |
| BG product free format text code listing | TRADE_BG_PRODUCT_FFT_LIST |

| Trade Finance Request | Consumer Service |
|--|--|
| BG product listing | TRADE_BG_PRODUCT_LIST |
| BG product read details | TRADE_BG_PRODUCT_READ |
| BG read details | TRADE_BG_READ |
| BG remarks | TRADE_BG_REMARK |
| BG summary details | TRADE_BG_SUMMARY_DETAILS |
| BG terms and conditions details | TRADE_BG_TERMSANDCONDITIONS_READ |
| BG type listing | TRADE_BG_TYPE_LIST |
| BG upload documents | TRADE_BG_UPLOAD_DOCUMENTS |
| BG version read | TRADE_BG_VERSION_READ |
| Bill charges list | TRADE_BILL_CHARGES_LIST |
| Bill charges simulation | TRADE_BILL_CHARGES_SIMULATION |
| Bill create | TRADE_BILL_CREATE |
| Bill discrepancy acceptance | TRADE_BILL_DISCREPANCY_CUST_ACCEPTANCE |
| Bill discrepancy listing | TRADE_BILL_DISCREPANCY_LIST |
| Bill discrepancy read details | TRADE_BILL_DISCREPANCY_READ |
| Bill swift and advice message read details | TRADE_BILL_LEVEL_SWIFT_ADVICE |
| Bill listing | TRADE_BILL_LIST |
| Bill product documents listing | TRADE_BILL_PRODUCT_DOCUMENTS |
| Bill product document clauses | TRADE_BILL_PRODUCT_DOCUMENTS_CLAUSES |
| Bill read detail | TRADE_BILL_READ |
| Bill settlement create | TRADE_BILL_SETTLEMENT_CREATE |
| Collection swift and advice message read details | TRADE_COLLECTION_ADVICE_SWIFT_READ |
| Collection charges list | TRADE_COLLECTION_CHARGES_LIST |
| Collection charges simulation | TRADE_COLLECTION_CHARGES_SIMULATION |

| Trade Finance Request | Consumer Service |
|--|-----------------------------------|
| Collection create | TRADE_COLLECTION_CREATE |
| Collection listing | TRADE_COLLECTION_LIST |
| Collection read details | TRADE_COLLECTION_READ |
| Contract free format text code list | TRADE_CONTRACT_FFT_LIST |
| Trade Customer clarification list | TRADE_CUSTOMER_CLARIFICATION_LIST |
| Trade detail summary snapshot | TRADE_DETAIL_SUMMARY_SNAPSHOT |
| Document category listing | TRADE_DOCUMENT_CATEGORY_LIST |
| Document content listing | TRADE_DOCUMENT_CONTENT_LIST |
| Document level clauses | TRADE_DOCUMENT_LEVEL_CLAUSES |
| Document listing | TRADE_DOCUMENT_LIST |
| Document category level doc type listing | TRADE_DOC_CATEGORY_LEVEL_DOC_TYPE |
| Trade equivalent amount | TRADE_EQUIVALENT_AMOUNT |
| Goods listing | TRADE_GOODS_LIST |
| Incoterm listing | TRADE_INCOTERM_LIST |
| LC additional conditions listing | TRADE_LC_ADDITIONAL_COND_LIST |
| LC advice and swift read details | TRADE_LC_ADVICE_SWIFT_READ |
| LC amend charges simulation | TRADE_LC_AMD_CHARGES_SIMULATION |
| LC amendment create | TRADE_LC_AMENDMENT_CREATE |
| LC amendment listing | TRADE_LC_AMENDMENT_LIST |
| LC amendment read details | TRADE_LC_AMENDMENT_READ |
| LC amend swift simulation | TRADE_LC_AMEND_SWIFT_SIMULATION |
| LC charges | TRADE_LC_CHARGES |
| LC charges simulation | TRADE_LC_CHARGES_SIMULATION |
| LC create | TRADE_LC_CREATE |
| LC availment listing | TRADE_LC_LEVEL_AVAILMENTS |

| Trade Finance Request | Consumer Service |
|-------------------------------------|-----------------------------------|
| LC listing | TRADE_LC_LIST |
| LC pending amendment listing | TRADE_LC_PENDING_AMENDMENT_LIST |
| LC pending amendment acceptance | TRADE_LC_PENDING_AMEND_ACCEPT |
| LC pending amendment rejection | TRADE_LC_PENDING_AMEND_REJECT |
| LC product currencies | TRADE_LC_PRODUCT_CURRENCIES |
| LC product documents | TRADE_LC_PRODUCT_DOCUMENTS |
| LC product document clauses | TRADE_LC_PRODUCT_DOCUMENT_CLAUSES |
| LC product listing | TRADE_LC_PRODUCT_LIST |
| LC product read details | TRADE_LC_PRODUCT_READ |
| LC read details | TRADE_LC_READ |
| LC swift simulation | TRADE_LC_SWIFT_SIMULATION |
| LC update | TRADE_LC_UPDATE |
| LC version read details | TRADE_LC_VERSION_READ |
| Trade read customer for bank code | TRADE_READ_CUST_FOR_BANK_CODE |
| SG advice and swift message details | TRADE_SG_ADVICE_SWIFT_READ |
| SG charges simulation | TRADE_SG_CHARGES_SIMULATION |
| SG create | TRADE_SG_CREATE |
| SG level charges | TRADE_SG_LEVEL_CHARGES |
| SG listing | TRADE_SG_LIST |
| SG product currency listing | TRADE_SG_PRODUCT_CURRENCY_LIST |
| SG product level documents | TRADE_SG_PRODUCT_LEVEL_DOCUMENTS |
| SG product listing | TRADE_SG_PRODUCT_LIST |
| SG product read details | TRADE_SG_PRODUCT_READ |
| SG read details | TRADE_SG_READ |
| Trade summary snapshot | TRADE_SUMMARY_SNAPSHOT |

| Trade Finance Request | Consumer Service |
|---|--|
| Export collection SWIFT message simulation | TRADE_COLLECTION_SWIFT_SIMULATION |
| SG charges | TRADE_SHIPPING_GUARANTEE_CHARGES_LIST |
| Document remarks | TRADE_UPDATE_REMARK |
| Application document download | TRADE_APPLICATION_DOCUMENT_DOWNLOAD |
| Transfer LC charges simulation | TRADE_TRANSFER_LC_CHARGES_SIMULATION |
| Transfer LC amendment | TRADE_TRANSFER_LC_AMENDMENT_CREATE |
| Corporate Deposit listing | TRADE_CORPORATE_DEPOSIT_LIST |
| Forex deal listing | TRADE_FOREX_DEAL_LIST |
| Amendable transferred LC listing | TRADE_TRANSFERRED_LC_AMEND_LIST |
| Transfer LC amendment SWIFT simulation | TRADE_TRANSFER_LC_AMEND_SWIFT_SIMULATION |
| Transfer LC SWIFT simulation | TRADE_TRANSFER_LC_SWIFT_SIMULATION |
| BG Cancellation charges simulation | TRADE_BG_CANCEL_CHARGES_SIMULATION |
| BG Cancellation SWIFT simulation | TRADE_BG_CANCEL_SWIFT_SIMULATION |
| BG Cancellation initiation | TRADE_BG_CANCELLATION_CREATE |
| Insurance policy list | TRADE_INSURANCE_POLICY_LIST |
| Customer direct dispatch flag read | TRADE_CUSTOMER_DIRECT_DISPATCH_READ |
| Export collection internal amendment | TRADE_EXPORT_COLLECTION_INTERNAL_AMENDMENT |
| Export collection modification charges simulation | TRADE_COLLECTION_MODIFY_CHARGES_SIMULATION |
| Export bill internal amendment | TRADE_EXPORT_BILL_INTERNAL_AMENDMENT |
| Export bill modification charges simulation | TRADE_BILL_MODIFY_CHARGES_SIMULATION |
| Charge and Taxes accounting entries | TRADE_TXN_CHARGE_TAXES_ACCOUNT_ENTRIES |
| Commission accounting entries | TRADE_TXN_COMMISSION_ACCOUNT_ENTRIES |

| Trade Finance Request | Consumer Service |
|--|--|
| Bill settlement charges simulation | TRADE_BILL_SETTLEMENT_CHARGES_SIMULATION |
| Collection settlement charges simulation | TRADE_COLLECTION_SETTLEMENT_CHARGES_SIMULATION |
| Import bill internal amendment | TRADE_IMPORT_BILL_INTERNAL_AMENDMENT |
| Import collection internal amendment | TRADE_IMPORT_COLLECTION_INTERNAL_AMENDMENT |
| Tracer charges simulation | TRADE_TRACER_CHARGE_SIMULATION |
| Tracer LC internal amendment | TRADE_TRACER_LC_INTERNAL_AMENDMENT |
| Tracer BG internal amendment | TRADE_TRACER_BG_INTERNAL_AMENDMENT |
| Collateral List | TRADE_COLLATERAL_LIST |
| Outstanding good list | TRADE_LC_OUTSTANDING_GOODS_LIST |
| Transfer LC product list | TRADE_TRANSFER_LC_PRODUCT_LIST |
| Customer application priority | TRADE_CUSTOMER_PRIORITY_READ |
| BG Pending claims list | TRADE_BG_CLAIM_SETTLEMENT_LIST |
| BG Claim charges simulation | TRADE_BG_CLAIM_SETTLEMENT_CHARGES_SIM |
| BG Claim Settlement | TRADE_BG_CLAIM_SETTLEMENT |

3.2 Oracle Banking Trade Finance Process Management (OBTFPM)

Following things need to be done to enable OBDX Trade Finance with OBTFPM

3.2.1 OBTF Prerequisite

Follow the steps from section **Oracle Banking Trade Finance (OBTF)**

Note: No need to restart the manage server though it is mentioned in section **OBRH Mandatory Executions**

3.2.2 **Mandatory Executions**

Execute the following script at OBDX database and restart the managed server.

```
../installables/db/OBTFFPM/DIGX_FW_CONFIG_ALL_O.sql
```

Note: '%ENTITY_ID%' should be replaced with entity identifier (For example 'OBDX_BU')

[Home](#)

4. Corporate Lending

4.1 Day One Executions

Following script needs to be executed post installation for Corporate Lending with OBCL 14.3.0.0.0 release

```
UPDATE digx_fw_config_all_o SET PROP_VALUE = CONCAT('OBCL14.1',(select
PROP_VALUE from digx_fw_config_all_o where PROP_ID like '&ENTITY_ID')) WHERE
PROP_ID LIKE '&ENTITY_ID';
```

Scripts mentioned in below path also needs to be executed:

installables/db/OBCL/version/DIGX_FW_CONFIG_ALL_O.sql

Note: '%ENTITY_ID%' should be replaced with entity identifier (For example 'OBDX_BU')

The list of OBCL and OBCLPM APIs that are integrated with OBDX using OBRH are as follows:

| Interface ID | Transaction Name | Description |
|--------------------------------|--------------------------|--|
| CORPORATE_BULK_LOAN_SETTLEMENT | Multiple Loan Settlement | This API is used to post single and bulk loan settlement to Loans Mid Office |
| LOAN_DISBURSEMENT_DETAILS | Disbursement Details | This API is used to fetch disbursement details from Loans Back Office |
| LOAN_SCHEDULE_DETAILS | Schedule Details | This API is used to fetch schedule details from Loans Back Office |
| LOAN_OUTSTANDING_DETAILS | Outstanding Details | This API is used to fetch outstanding details from Loans Back Office |
| LOAN_DETAILS | Loan Details | This API is used to fetch loan details from Loans Back Office |
| LOAN_ACCOUNT_LIST | Loan Account List | This API is used to fetch list of accounts from Loans Back Office |

| Interface ID | Transaction Name | Description |
|---|----------------------------|---|
| LOAN_RATEREVISION_DETAILS | Rate Revision Details | This API is used to fetch rate revision details from Loans Back Office |
| LOAN_FULL_OUTSTANDING_DETAILS | Full Outstanding Details | This API is used to fetch full outstanding details from Loans Back Office |
| CORPORATE_LOAN_SWIFT_MESSAGE_OR_ADVICES | Swift Message or Advices | This API is used to fetch corporate loan swift message and advices from Loans Back Office |
| CORPORATE_LOAN_SETTLEMENT_SIMULATION | Loan Settlement Simulation | This API is used to fetch corporate loan settlement simulation from Loans Back Office |
| CORPORATE_LOAN_SETTLEMENT | Loan Settlement | This API is used to post corporate loan settlement from Loans Back Office |
| LOAN_PRODUCT_PROCESSING_CHARGES | Processing Charges | This API is used to fetch processing charges of a product from Loans Back Office |
| LOAN_DRAWDOWN_APPLICATION | Drawdown Application | This API is used to post drawdown application to Loans Mid Office |
| LOAN_APP_FETCH_APPLICATION_STATUS | Application Status | This API is used to fetch status of a drawdown application from Loans Mid Office |
| LOAN_APP_PRODUCT_DETAILS | Product Details | This API is used to fetch product details from Loans Mid Office |
| LOAN_APP_PRODUCT_LIST | Product List | This API is used to fetch product list from Loans Mid Office |

| Interface ID | Transaction Name | Description |
|--------------------------|------------------|---|
| LOAN_APP_PRODUCT_SEGMENT | Product Segment | This API is used to fetch product segment from Loans Mid Office |
| LOAN_APP_PURPOSE_DETAILS | Purpose List | This API is used to fetch purpose list from Loans Mid Office |

[Home](#)

5. Supply Chain Finance

5.1 Non Customer Onboarding using chaining

Now in the standard scenario, the core system contains the Customer data and the OBSCF mid office system contains the Non Customer data. Thus in order to onboard a noncustomer (give channel access) the system needs to inquire in OBSCF mid office.

But till now the system was inquiring only in Core system, which we still need for the onboarding of customers. Thus a concept of chaining is introduced where for a given corporate, the system will first inquire in Core system and if found then the given corporate is a customer but if not found then the system will inquire in OBSCF mid office system and if found there then the given corporate is a noncustomer.

Now, the chaining is not only implemented for 2 levels (calling only 2 systems) but it can be implemented for n levels. Also there is a provision to break a chain at any level or if there is a case that there is an overridden adapter to call a common system containing both customers and noncustomers and not want to call core system and mid office system adapters i.e. not implement/require chaining at all, then this is also possible.

For detail explanation of Chaining, how it works, chaining in case of overridden adapters and many more please refer **Chaining Section in Extensibility Document**.

Now below are the scenarios of how chaining will be used for Non Customer Onboarding in case of different possible implementations at Bank.

Considering, IPartyAdapter has 3 implementation

i1 - PartyAdapter(UBS), i2 - PartyAdapter(ASP) & i3 - PartyAdapter(TP)

Case 1:

Bank has both UBS core entity and ASP mid office as well (OBASP).

In this case, the entry for UBS core entity in DIGX_FW_CONFIG_ALL_O will be like (Assuming OBDX_BU is the determinant value for UBS core entity)

OBDX_BU | UBS, ASP, TP

Note: Here entry of TP might be for other mid offices system but not for UBS Core and OBSCF mid office as both are present with the bank according to the case.

Thus in case of chaining, it will first inquire in "i1 Adapter", if found then it will stop and return the result. If not found then it will inquire in "i2 Adapter", if found then it will stop and return the result. If not found then it will inquire in "i3 Adapter", where there are maximum chances that it won't be found because of above note. Thus finally after not able to find in "i3 Adapter", it will throw the error like it used to throw before chaining when not found in core system.

Case 2:

Bank has UBS core entity but ASP mid office is Third Party.

In this case, the entry for UBS core entity in DIGX_FW_CONFIG_ALL_O will be like (Assuming OBDX_BU is the determinant value for UBS core entity)

OBDX_BU | UBS, TP

Thus in case of chaining, it will first inquire in "i1 Adapter", if found then it will stop and return the result. If not found then it will inquire in "i3 Adapter". Now in case of i3, it will push the request in JMS queue which will be read by one of the middleware implementations and sent to the actual Third party mid office of ASP. If found then the result will returned but if not then it will throw the error like it used to throw before chaining when not found in core system.

** In this case, all the other functionalities of ASP mid office like Onboarding Associated Party will also be achieved by using Third Party implementation of ASP.

Case 3:

Bank has Third party core entity and Third Party ASP mid office

In this case, the entry for Third Party core entity in DIGX_FW_CONFIG_ALL_O will be like (Assuming OBDXBU1 is the determinant value for Third Party core entity)

OBDXBU1 | TP

Thus, here there is no scenarios of chaining as always only "i3 Adapter" will be picked. Now in case of i3, it will push the request in JMS queue which will be read by one of the middleware implementations and sent to the actual single system (like common core). Now that single system can have the logic to check the party in core system & ASP system if required.

** In this case, all the other functionalities of ASP mid office like Onboarding Associated Party will also be achieved by using Third Party implementation of ASP.

Case 4:

Bank has Third Party core entity but ASP mid office is of OBASP

In this case, the entry for Third Party core entity in DIGX_FW_CONFIG_ALL_O will be like (Assuming OBDXBU1 is the determinant value for Third Party core entity)

OBDXBU1 | TP, ASP

Thus in case of chaining, it will first inquire in "i3 Adapter". Now in case of i3, it will push the request in JMS queue which will be read by one of the middleware implementations and sent to the actual Third party core system. If found then it will stop and return the result. If not found then it will inquire in "i2 Adapter". If found then the result will be returned but if not then then it will throw the error like it used to throw before chaining when not found in core system.

** In this case, all the other functionalities of ASP mid office like Onboarding Associated Party should only be achieved by ASP host implementation (one that is qualified with OBASP). For that, **we need to override the scripts of host adapter in DIGX_FW_CONFIG_ALL_O such that for ASP functionalities it will always pick the ASP adapter and for other common functionalities like Get Non Customer party, chaining will be applied as explained above.**

We need to execute below script to fetch mid office token required for Purchase Order File Upload.

```
INSERT INTO DIGX_FW_CONFIG_OUT_RS_CFG_B (SERVICE_ID, CONTEXT_URL,
SERVICE_URL, REQUEST_MEDIA_TYPE, RESPONSE_MEDIA_TYPE, AUTHENTICATION,
AUTH_TYPE, CREDENTIAL_STORE_TYPE, CREDENTIAL_STORE_KEY, CREATION_DATE,
LAST_UPDATED_DATE) VALUES ('tokenOBSCF144',
'http://${OBSCF_HOST_IP}:${OBSCF_HOST_PORT}','api-gateway/platojwtauth',
'application/json', 'application/json', 'N', 'Bearer', 'credential_impl', 'OBSCF_14.4', sysdate,
sysdate);
```

5.2 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of OBSCF installation.

For OBDX and mid Office OBSCF integration using OBRH the following configurations need to be done.

1. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for mid-office product OBSCF (Oracle Banking Supply Chain Finance) end-points configured in OBRH is "OBSCF"
2. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:

[/Installables/Modules/OBSCF/DIGX_FW_CONFIG_ALL_O.sql](#)

Note: '%ENTITY_ID%' should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then '%ENTITY_ID%' should be replaced by 'OBDX_BU').

3. The list of OBSCF APIs that are integrated with OBDX using OBRH is as follows:

| OBRH Consumer Service Name | Transaction Name | Description |
|----------------------------|-------------------------|--|
| SCF_FINANCE_CHARGE | View/Edit Invoice | Fetches the list of finance charges. |
| SCF_PURCHASE_ORDER_CREATE | Purchase Order Creation | This API is used to create purchase orders. |
| SCF_PURCHASE_ORDER_UPDATE | View Purchase Orders | This API is used to modify purchase order details. |

| OBRH Consumer Service Name | Transaction Name | Description |
|-----------------------------------|------------------------------|--|
| SCF_PURCHASE_ORDER_ACCEPT | Accept/Reject Purchase Order | This API is used to accept purchase orders. |
| SCF_PURCHASE_ORDER_REJECT | Accept/Reject Purchase Order | This API is used to reject purchase orders. |
| SCF_PURCHASE_ORDER_CANCEL | View Purchase Orders | This API is used to cancel purchase orders. |
| SCF_PURCHASE_ORDER_LIST | View Purchase Orders | Fetches the list of purchase orders. |
| SCF_PURCHASE_ORDER_READ | View Purchase Orders | Fetches purchase order details. |
| SCF_LINKED_PO_LIST | View Finance Details | Fetches Purchase Orders linked to a finance. |
| SCF_PO_FINANCE_CREATE | Request Finance | This API is used to request finance on purchase order(s). |
| SCF_FINANCE_LIMITS_LIST | View Limits | Fetches the list of finance limits. |
| SCF_MAIN_LIST | View Limits | Fetches the list of supply chain finance maintenances for a key. |
| SCF_PROGRAM_LIST | View/Edit Program | Fetches the list of programs. |
| SCF_PROGRAM_READ | View/Edit Program | Fetches program details. |
| SCF_PROGRAMPRODUCT_LIST | Create Program | Fetches the list of Program products. |
| SCF_PROGRAMPRODUCT_READ | Create Program | Fetches program product details. |
| SCF_PROGRAM_CREATE | Create Program | Creates Program |
| SCF_PROGRAM_UPDATE | Edit Program | Edit program details |
| SCF_LINKED_FINANCE_LIST | View Invoice Details | Fetches finances linked to an invoice. |
| SCF_FINANCE_CREATE | Request Finance | This API is used to request finance on invoice(s). |

| OBRH Consumer Service Name | Transaction Name | Description |
|-----------------------------|-----------------------------------|---|
| SCF_FINANCE_REPAYMENT | Repay Finance | Initiates request to repay finance. |
| SCF_FINANCE_SETTLEMENT | View Finance Details | Fetches finance settlement details |
| SCF_FINANCE_READ | View Finance Details | Fetches finance details |
| SCF_FINANCE_LIST | View Finance | Fetches list of finances |
| SCF_LINKED_INVOICE_LIST | View Finance Details | Fetches invoices linked to a finance. |
| SCF_LINK_INVOICE_TO_PROGRAM | Link Invoice To Program | Links Invoice(s) to Program |
| SCF_DISCOUNT_OFFER_CREATE | Create Discount Offer | Creates Discount Offer |
| SCF_DISCOUNTOFFER_LIST | View Discount Offer | Fetches list of discount offers |
| SCF_DISCOUNT_OFFER_READ | View Discount Offer Details | Fetch discount offer details |
| SCF_LINKED_DISCOUNT_OFFERS | View Receivables/Payables Details | Fetches list of offers linked to an invoice |
| SCF_CHARGE_CALCULATION | Manage Receivables/Payables | Fetches applicable discount on an invoice |

This completes the entire configuration needed for consuming OBSCF APIs in OBDX through OBRH.

- In addition to the above, below script needs to be executed for successful creation of Discount Offer.

```
insert into DIGX_FW_TAXONOMY_DATA_TYPE_MAP
(ID,TYPE,DATATYPEID,MINLENGTH,MAXLENGTH,MANDATORY,ERRORCODE,LENGTH_ER
ROCODE,MANDATORY_ERRORCODE,CREATION_DATE,CREATED_BY,LAST_UPDATED_D
ATE,LAST_UPDATED_BY,OBJECT_VERSION_NUMBER) values
```

```
('com.ofss.digx.app.scf.dto.discountoffer.DiscountOfferRequestDTO.discountOffer.invoices.indicator','CLASS','FREETEXT',null,null,'N',null,null,null,sysdate,'ofssuser',sysdate,'ofssuser',1);
```

6. Receivables/Payables Management

6.1 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of either OBCM or OBSCF installation.

For OBDX and mid Office OBSCFCM integration using OBRH the following configurations need to be done.

1. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for mid-office product OBSCFCM (Oracle Banking Cash Management or Oracle Banking Supply Chain Finance) end-points configured in OBRH is “ASP”, “INV” and “SCFCM”.
2. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:

/Installables/db/OBSCFCM/version/DIGX_FW_CONFIG_ALL_O.sql

Note: ‘%ENTITY_ID%’ should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then ‘%ENTITY_ID%’ should be replaced by ‘OBDX_BU’).

3. The list of OBSCFCM APIs that are integrated with OBDX using OBRH is as follows:

| OBRH Consumer Service Name | Transaction Name | Description |
|----------------------------------|---|---|
| ASP_ASSOCIATEDPARTY_CREATE | Onboard Associated Party | This API is used to onboard an associated party. |
| ASP_ASSOCIATEDPARTY_LIST | View Associated Parties | Fetches the list of associated parties. |
| ASP_ASSOCIATEDPARTY_READ | View Associated Parties | Fetches associated party details. |
| ASP_ASSOCIATEDPARTY_UPDATE | Upload KYC document for Non-Customer | This API is used to update the document Id for a new associated party that is not a customer of a bank. |
| ASP_ASSOCIATEDPARTY_RELATIONSHIP | Fetch Buyer-Supplier Relationship details | This interface is used to fetch details of buyer-supplier relationship |
| INV_INVOICES_CREATE | Create Receivables/Payables | This API is used to create invoices. |

| OBRH Consumer Service Name | Transaction Name | Description |
|-------------------------------------|--------------------------------|--|
| INV_INVOICE_LIST | View/Edit Receivables/Payables | This API is used to fetch invoices |
| INV_INVOICE_READ | View/Edit Receivables/Payables | This API is used to fetch invoice details |
| INV_INVOICES_UPDATE_STATUS | Manage Receivables/Payables | This API allows a user to perform various operations on invoices like Edit, Cancel, Accept, Raise Dispute, Resolve Dispute etc |
| INV_INVOICES_DISPUTE_REASON | Manage Receivables/Payables | This API fetches list of dispute reasons required to raise dispute on an invoice. |
| INV_LIST_COMMODITIES | Create Receivables/Payables | This API fetches list of supplier based commodities. |
| INV_CREDIT_NOTE_CREATE | Create Credit Note | This API is used to create credit notes |
| INV_CREDIT_NOTE_LIST | View Credit Note | This API is used to fetch credit notes |
| INV_CREDIT_NOTE_READ | View Credit Note | This API is used to fetch credit note details |
| SCFCM_PARAMS_LIST | Onboard Associated Party | This API fetches application params. |
| SCFCM_PAYMENTS_LIST | View Payments | This API is used to fetch the list of payments. |
| SCFCM_PAYMENT_READ | View Payment Details | This API is used to fetch payment details. |
| SCFCM_MANUAL_RECONCILIATION | Manual Reconciliation | This API is used to manually reconcile cashflows/invoices with payments |
| SCFCM_LIST_RECONCILED_TRANS ACTIONS | De-Reconciliation | This API is used to fetch the list of reconciled cashflows/invoices. |

| OBRH Consumer Service Name | Transaction Name | Description |
|--------------------------------------|----------------------------------|--|
| SCFCM_DERECONCILE | De-Reconciliation | This API is used to de-reconcile already reconciled cashflows/invoices. |
| SCFCM_RECONCILIATION_RULE_LIST | View/Edit Reconciliation Rules | Fetches list of reconciliation rules maintained for a party. |
| SCFCM_RECONCILIATION_RULE_CREATE | Create Reconciliation Rule | This API is used to create reconciliation/allocation rule for a party. |
| SCFCM_RECONCILIATION_RULE_UPDATE | Edit Reconciliation Rule | This API is used to modify reconciliation/allocation rule details for a party. |
| SCFCM_RECONCILIATION_RULE_READ | View Reconciliation Rule details | Fetches reconciliation rule details. |
| SCFCM_LIST_RECONCILIATION_CATEGORIES | View/Edit Reconciliation Rules | Fetches reconciliation categories |
| SCFCM_LIST_RECONCILIATION_ATTRIBUTES | View/Edit Reconciliation Rules | Fetches reconciliation attributes |
| SCFCM_LIST_ALLOCATED_TRANSACTIONS | View Payment Details | This API is used to fetch allocation details of a payment. |
| SCFCM_LIST_ALLOCATION_ACCOUNTS | Manual Allocation | This API is used to fetch virtual accounts which can be further allocated to payments. |
| SCFCM_MANUAL_ALLOCATION | Manual Allocation | This API is used to manually allocate payments to virtual account. |

This completes the entire configuration needed for consuming OBSCFCM APIs in OBDX through OBRH.

We need to execute below script to fetch mid office token required for fetching associated party information for non-customer.

```
INSERT INTO DIGX_FW_CONFIG_OUT_RS_CFG_B (SERVICE_ID, CONTEXT_URL,  
SERVICE_URL, REQUEST_MEDIA_TYPE, RESPONSE_MEDIA_TYPE, AUTHENTICATION,  
AUTH_TYPE, CREDENTIAL_STORE_TYPE, CREDENTIAL_STORE_KEY, CREATION_DATE,  
LAST_UPDATED_DATE) VALUES ('tokenASP144',  
'http://{ASP_HOST_IP}:{ASP_HOST_PORT}', 'api-gateway/platojwtauth',  
'application/json', 'application/json', 'N', 'Bearer', 'credential_impl', 'ASP_14.4', sysdate,  
sysdate);
```

We need to execute below script to fetch mid office token required for Invoice/Debit-Credit Note File Upload.

```
INSERT INTO DIGX_FW_CONFIG_OUT_RS_CFG_B (SERVICE_ID, CONTEXT_URL,  
SERVICE_URL, REQUEST_MEDIA_TYPE, RESPONSE_MEDIA_TYPE, AUTHENTICATION,  
AUTH_TYPE, CREDENTIAL_STORE_TYPE, CREDENTIAL_STORE_KEY, CREATION_DATE,  
LAST_UPDATED_DATE) VALUES ('tokenINV144',  
'http://{INV_HOST_IP}:{INV_HOST_PORT}', 'api-gateway/platojwtauth', 'application/json',  
'application/json', 'N', 'Bearer', 'credential_impl', 'INV_14.4', sysdate, sysdate);
```

[Home](#)

7. Virtual Account Management

7.1 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of OBVAM installation.

For OBDX and OBVAM integration using OBRH the following configurations need to be done.

1. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for mid-office product OBVAM (Oracle Banking Virtual Account Management) end-points configured in OBRH is “**OBVAM**” (this step is not required to be repeated after each patch-set).
2. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:
 - a. /installables/db/OBVAM/version/DIGX_FW_CONFIG_ALL_O.sql (part of base installer)

Note: ‘%ENTITY_ID%’ in the above scripts should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then ‘%ENTITY_ID%’ should be replaced by ‘OBDX_BU’).

All the OBVAM APIs consumed from OBDX are via OBRH. List is as follows:-

| OBRH Consumer Service name | Transaction name |
|------------------------------|-------------------------------------|
| abortVirtualAccountClosure | Terminate Virtual Account Closure |
| closeRemittance | Close Remittance ID |
| closeVA | Close Virtual Account |
| closeVirtualAccountStructure | Close Virtual Account Structure |
| createCreditLineLinkage | Create Internal Credit Line Linkage |
| createGeneralRates | Add Generate Rates |
| createInternalCreditLine | Create Internal Credit Line |
| createInternalTransfer | Move Money |
| createRemittance | Create Remittance ID |
| createSpecialRates | Add Special Rates |
| createVirtualAccount | Create Virtual Account |

| | |
|-----------------------------------|--|
| createVirtualAccountStructure | Create Virtual Account Structure |
| createVirtualEntity | Create Virtual Entity |
| createVirtualMultiCurrencyAccount | Create Virtual Multi Currency Account |
| deleteCreditLineLinkage | Delete Internal Credit Line Linkage |
| deleteInternalCreditLine | Delete Internal Credit Line |
| deleteVirtualEntity | Close Virtual Entity |
| deleteVirtualMultiCurrencyAccount | Close Virtual Multi Currency Account |
| downloadVASChildAccounts | Download Virtual Account Structure |
| editVirtualMultiCurrencyAccount | Edit Virtual Multi Currency Account |
| fetchBranchDateByBranchCode | Fetch Branch Date |
| fetchDefaultInterestRates | Fetch Default Rates/UDEs |
| fetchDistinctCurrencies | Fetch Distinct Currencies (Projection) |
| fetchEntityBankParameters | Fetch Entity Bank Parameters |
| fetchInterestHistory | Fetch Interest Rates History |
| fetchRateCodes | Fetch Rate Codes |
| fetchRates | Fetch Interest Rates (General/Special) |
| fetchRemittanceList | Fetch Remittance IDs |
| fetchVAMCountry | Fetch Countries (CMC) |
| fetchVAMCurrency | Fetch Currencies (CMC) |
| fetchVAMEnabledAccountsForParty | Fetch VAM Enabled Real Accounts |
| fetchVAStatement | Fetch Virtual Account Transactions |
| fetchVAforLinkage | Fetch Virtual Accounts for Credit Line Linkage |
| fetchVAwithBalance | Fetch Virtual Account with Balance and Structure |
| fetchValueDatedBalances | Fetch Value Dated Balances |
| fetchVamChargeDefinitionList | Fetch Charge Definitions |
| fetchVamChargeList | Fetch Charges |

| | |
|--|--|
| fetchVirtualAccountBalances | Fetch Virtual Account Balance |
| fetchVirtualAccountBranches | Fetch Virtual Account Branches |
| fetchVirtualAccountList | Fetch Virtual Accounts |
| fetchVirtualMultiCurrencyAccount | Fetch Virtual Multi Currency Accounts |
| fetchVirtualProduct | Fetch Virtual Account Products |
| getAccruedAmountForVirtualAccount | Fetch Interest Accrued Amount for Virtual Account |
| getChildAccountsForVirtualStructure | Fetch Child Accounts for Parent Account in Virtual Account Structure |
| getIdentificationTypesForVirtualEntity | Fetch Identification Types for Virtual Entity |
| getLineAccountUtilization | Fetch Internal Credit Line Utilization for Virtual Account |
| getLineUtilization | Fetch Internal Credit Line Utilization |
| getRemitterCountForVI | Fetch Remittance ID count for Virtual Identifier |
| getUnmappedVirtualAccounts | Fetch Virtual Accounts eligible for adding in Structure |
| getVirtualAccountClosureStatusDetails | Fetch Virtual Account Closure Status details |
| getVirtualAccountStructuresByCustomer | Fetch Virtual Account Structures |
| listCreditLineLinkage | Fetch Internal Credit Line Linkages |
| listInternalCreditLine | Fetch Internal Credit Lines |
| listVirtualAccWithStructureCode | Fetch Virtual Accounts part of a Structure |
| listVirtualEntity | Fetch Virtual Entities |
| listVirtualIdentifier | Fetch Virtual Identifiers |
| modifyVirtualAccountStructure | Edit Virtual Account Structure |
| readInternalCreditLine | Fetch Internal Credit Line details |
| readVirtualAccount | Fetch Virtual Account details |
| readVirtualEntity | Fetch Virtual Entity details |
| readVirtualMultiCurrencyAccount | Fetch Virtual Multi Currency details |

| | |
|------------------------------------|--|
| reopenRemittance | Reopen Remittance ID |
| reopeningClosedVirtualAccount | Reopen Virtual Account |
| retryVirtualAccountClosure | Retry Virtual Account Closure |
| transferVirtualAccount | Fetch Transfer Virtual Accounts for Closure |
| updateCreditLineLinkage | Edit Internal Credit Line Linkage |
| updateInternalCreditLine | Edit Internal Credit Line |
| updateRemittance | Edit Remittance ID |
| updateVirtualAccount | Edit Virtual Account |
| updateVirtualEntity | Edit Virtual Entity |
| vaForClosure | Fetch Virtual Accounts eligible for Closure |
| vaForClosureStatus | Fetch Virtual Accounts initiated for Closure along with Status |
| vamFetchAdhocCAMTReport | Fetch Adhoc CAMT Statement |
| vamFetchAdhocMTReport | Fetch Adhoc MT Statement |
| vamFetchAdhocPDFReport | Fetch Adhoc PDF Statement |
| vamFetchPreGenReport | Fetch Pre-generated Statement |
| virtualAccountsforInternalTransfer | Fetch Virtual Accounts for Real Account |
| fetchVirtualAccountRestrictions | Fetch Virtual Account Restrictions |
| maintainVirtualAccountRestriction | Edit Virtual Account Restrictions |

This completes the entire configuration needed for consuming OBVAM APIs in OBDX through OBRH.

7.2 Verify System Configurations

Following script helps in listing the VAM specific System Configurations:-

```
SELECT prop_id as PROPERTY_IN_DATABASE, NVL(SUBSTR(t.UI_definition,
INSTR(t.UI_definition, "title")+9, INSTR(t.UI_definition, ",", "-11), t.UI_definition) AS
TITLE_ON_SCREEN, t.prop_value FROM digx_fw_config_var_b t where prop_id like '%VAM%'
and module = 'OTHERMODULE' and determinant_value = '*'; -- Please enter correct determinant
value
```

Ensure correct values are maintained against the above properties.

This maintenance can be done from the “System Configuration” admin screen or directly in DB schema.

Do ensure to follow the **User Credential Configuration** section in this document for configuring Connector Credential store. The OBVAM credentials required to generate token for making outbound REST API calls from OBDX to OBVAM needs to be stored here (not required to be repeated after each patch-set).

7.3 Enumerations

Following VAM related enumerations are used in OBDX. They are used to fetch the values on the OBDX UI.

- a. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getIdentificationTypes';*
- b. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getCorporateTypes';*

The mapping of these OBDX values to the corresponding OBVAM values can be found in next section.

7.4 Adapter Properties

Certain fields (Eg: Enumerations, Status etc) can have different values in OBDX as compared to OBVAM.

The mapping of all such values between OBDX and OBVAM can be found/maintained using the below script:-

```
select * from DIGX_FW_CONFIG_ADAPTER_PROP_B where host_id = 'OBVAM';
```

[Home](#)

8. Cash Management System

8.1 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of OBCM installation.

For OBDX and mid Office OBCM integration using OBRH the following configurations need to be done.

1. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for mid-office product OBCM (Oracle Banking Cash Management) end-points configured in OBRH is “OBCM”
2. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:

/Installables/db/OBCM/version/DIGX_FW_CONFIG_ALL_O.sql

Note: ‘%ENTITY_ID%’ should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then ‘%ENTITY_ID%’ should be replaced by ‘OBDX_BU’).

3. The list of OBCM APIs that are integrated with OBDX using OBRH is as follows:

| OBRH Consumer Service Name | Transaction Name | Description |
|---------------------------------|--------------------------------------|--|
| CMS_CASHFLOW_TRANSACTION_CREATE | Create Expected Cash Flow | This API is used to create expected cashflows. |
| CMS_CASHFLOW_TRANSACTION_LIST | View/Edit Expected Cash Flow Details | Fetches the list of cashflows. |
| CMS_CASHFLOW_TRANSACTION_READ | View/Edit Expected Cash Flow Details | Fetches Cashflow Details |
| CMS_CASHFLOW_TRANSACTION_UPDATE | View/Edit Expected Cash Flow Details | This API is used to modify expected cashflow details. |
| CMS_CASHFLOW_FETCH | Cashflow Forecasting | This API is used to fetch cashflow forecasting data. |
| CMS_CCM_LIST | Cash Deposits | This API is used to fetch collection maintenance details of a cash management. |
| CMS_DIVISION_CODE_LIST | Cash & Cheque Deposits | This API is used to fetch division code details of a given party. |

| OBRH Consumer Service Name | Transaction Name | Description |
|-----------------------------------|------------------------------|---|
| CMS_CASH_DEPOSITS_CREATE | Cash Deposits | This API is used to create multiple cash deposits for a party. |
| CMS_CASH_DEPOSITS_LIST | View Cash Deposits | This API is used to fetch cash collections. |
| CMS_CASH_DEPOSIT_READ | View Cash Deposit Details | This API is used to fetch cash collection details. |
| CMS_CHEQUE_LIST | View Cheque Deposits | This API is used to fetch cheque collections. |
| CMS_CHEQUE_READ | View Cheque Deposit Details | This API is used to fetch cheque collection details. |
| CMS_BRANCH_DENOMINATION_LIST | Cash Deposits | This API is used to fetch denomination details for a particular branch. |
| CMS_BRANCH_CODE_LIST | Cash & Cheque Deposits | This API is used to fetch all the branch. |
| CMS_CASH_DEPOSIT_VALIDATE | Cash Deposits | This API is used to validate cash/cheque number uniqueness. |
| CMS_CMM_LIST | Cash Deposits | This API is used to fetch maintenance details of a cash management. |
| CMS_CASH_WITHDRAWAL_LIST | View Cash withdrawal | This API is used to fetch cash withdrawal collections. |
| CMS_CASH_WITHDRAWAL_READ | View Cash withdrawal details | This API is used to fetch cash withdrawal collection details. |
| CMS_CASH_WITHDRAWAL_CREATE | Create Cash withdrawal | This API is used to create cash withdrawal for a party. |
| CMS_CHEQUE_DEPOSITS_CREATE | Cheque Deposits | This API is used to create multiple cheque deposits for a party. |
| CMS_CASHFLOW_CODE_LIST | Cashflow Forecasting | This API is used to fetch cash flow code details. |
| CMS_ROUTING_LIST | Cheque Deposits | This API is used to fetch routing details. |

| OBRH Consumer Service Name | Transaction Name | Description |
|----------------------------|-----------------------------|--|
| CMS_COLLECTION_LIST | Overview Collection summary | This API is use to fetch collection summary details for cash/cheque & cash withdraw. |
| CMS_PDM_LIST | Pick up & delivery | This API is use to fetch pickup and delivery details. |

This completes the entire configuration needed for consuming OBCM APIs in OBDX through OBRH.

We need to execute below script to fetch mid office token required for Cashflow/Payment File Upload.

```
INSERT INTO DIGX_FW_CONFIG_OUT_RS_CFG_B (SERVICE_ID, CONTEXT_URL,
SERVICE_URL, REQUEST_MEDIA_TYPE, RESPONSE_MEDIA_TYPE, AUTHENTICATION,
AUTH_TYPE, CREDENTIAL_STORE_TYPE, CREDENTIAL_STORE_KEY, CREATION_DATE,
LAST_UPDATED_DATE) VALUES ('tokenOBCM144',
'http://{OBCM_HOST_IP}:{OBCM_HOST_PORT}', 'api-gateway/platojwtauth',
'application/json', 'application/json', 'N', 'Bearer', 'credential_impl', 'OBCM_14.4', sysdate,
sysdate);
```

[Home](#)

9. Credit Facility Management

9.1 Core as Third Party

During Bank Implementation, in the case that the core system is Third Party and Mid-Office is OBCFPM and ELCM (Oracle Banking Credit Facility Management), then the entry in DIGX_FW_CONFIG_ALL_O will be: Determinant value for Third Party Entity | TP1.0, OBCFPM14.4,OBCFPM14.3, ELCM14.4,ELCM14.3

For example, if the determinant value for the Third Party Entity is OBDXBU1 then the entry will look like: OBDXBU1 | TP1.0, OBCFPM14.4,OBCFPM14.3, ELCM14.4,ELCM14.3

9.2 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of OBCFPM installation.

For OBDX and mid Office OBCFPM integration using OBRH the following configurations need to be done.

4. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for mid-office product OBCFPM (Oracle Credit facility Management) end-points configured in OBRH is “OBCFPM”
5. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:

/installables/db/OBCFPM/DIGX_FW_CONFIG_ALL_O.sql

/installables/db/ELCM/DIGX_FW_CONFIG_ALL_O.sql

Note: ‘%ENTITY_ID%’ should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then ‘%ENTITY_ID%’ should be replaced by ‘OBDX_BU’).

6. The list of OBCFPM and ELCM APIs that are integrated with OBDX using OBRH is as follows:

| Interface Id | Transaction Name | Description |
|--------------------------|--------------------------|--|
| CF_LIABILITY_READ | Credit Facility Overview | This API is used to fetch the liability details of particular liability. |
| CF_LIABILITY_LIST | Credit Facility Overview | This API is used to fetch the liability details. |
| CF_COLLATERALGROU P_READ | Collateral Summary | This API is used to fetch the collateral group details of particular collateral. |

| Interface Id | Transaction Name | Description |
|--------------------------------|-----------------------|--|
| CF_COLLATERALGROUP_LIST | Collateral Summary | This API is used to fetch the collateral group details. |
| CF_COLLATERAL_LIST | Collateral Summary | This API is used to fetch the collateral details. |
| CF_COLLATERAL_READ | Collateral Summary | This API is used to fetch the collateral details of particular collateral. |
| CF_FACILITY_LIST | Facility Summary | This API is used to fetch the facility details |
| CF_FACILITY_CATEGORIES_SERVICE | Facility Summary | This API is used to fetch the facility category of particular facility. |
| CF_FACILITY_UTILIZATION | Facility Details | This API is used to fetch the facility history details |
| CF_COLLATERALTYPES_LIST | Collateral Evaluation | This API is used to fetch the collateral types. |
| CF_FACILITYCATEGORY_LIST | Apply new Facility | This API is used to fetch the facility category. |
| CF_FACILITYCATEGORY_LIST | Apply new Facility | This API is used to fetch the facility category. |
| CF_DOCUMENT_READ | Apply new Facility | This API is used to fetch the document. |
| CF_FETCH_APPLICATION_STATUS | Apply new Facility | This API is used to fetch the application status. |
| CF_FACILITY_UPDATE | Apply new Facility | This API is used to create and update facility. |

| Interface Id | Transaction Name | Description |
|-----------------------------|--------------------------|--|
| CF_COLLATERAL_OFFER | Application Tracker | This API is used to accept or reject applications. |
| CF_EVALUATE_COLLATERAL | Collateral Evaluation | This API is used to evaluate collateral. |
| CF_REVALUATE_COLLATERAL | Collateral Revaluation | This API is used to revalue collateral. |
| CF_COLLATERAL_READ_MULTIPLE | Collateral Read Multiple | Collateral Read for multiple collateral details |

[Home](#)

10. Liquidity Management

10.1 OBRH Integration

During Bank Implementation, assuming OBRH is installed and configured as part of OBLM installation.

For OBDX and OBLM integration using OBRH the following configurations need to be done.

1. Carry out all the steps mentioned in **OBRH Integration Configuration** section in this document. The service provider for product OBLM (Oracle Banking Liquidity Management) end-points configured in OBRH is “**OBLM**” (this step is not required to be repeated after each patch-set).
2. After all the above steps are completed, user needs to execute some scripts for the host APIs that are to be consumed via OBRH. These are the scripts to pick the third party adapter implementation instead of the host specific implementations as well as to call the OBRH end-point for the configured interfaces. The scripts are available at the following location:
 - a. /installables/db/OBLM/version/DIGX_FW_CONFIG_ALL_O.sql (part of base installer)

Note: ‘%ENTITY_ID%’ in the above scripts should be replaced with the entity identifier (For Example if the entity during implementation is OBDX_BU then ‘%ENTITY_ID%’ should be replaced by ‘OBDX_BU’).

All the OBLM APIs consumed from OBDX are via OBRH. List is as follows:

| OBRH Consumer Service Name | Transaction Name |
|------------------------------------|--|
| createLMStructure | Create Structure |
| editLMStructure | Edit Structure |
| executeLMPoolStructure | Execute Pool Structure |
| executeLMSweepStructure | Execute Sweep Structure |
| fetchLMAccounts | Fetch Liquidity enabled Accounts for Primary as well as Linked Customers |
| fetchLMAccountsByPartyId | Fetch Liquidity enabled Accounts for Customer |
| fetchLMAccountsWithStructure | Fetch Accounts participating in multiple Structures |
| fetchLMAccountsWithlinkedStructure | Fetch Structure details for list of Accounts |
| fetchLMBranches | Fetch Branches |

| | |
|-------------------------------|--|
| fetchLMChargeDefinitionList | Fetch Charge Definitions |
| fetchLMChargeList | Fetch Charges |
| fetchLMCurrency | Fetch Currencies |
| fetchLMFrequencies | Fetch Frequencies |
| fetchLMInstruction | Fetch Instructions |
| fetchPoolLogs | Fetch Pool Logs |
| fetchSweepLogs | Fetch Sweep Logs |
| fetchUpcomingSweepLogs | Fetch Upcoming Sweep Logs |
| listStructurePriorities | Fetch Structure Priorities |
| partyHierarchyList | Fetch Linked Customers Hierarchy |
| readLMStructure | Fetch Structure details |
| validateLMStructure | Validate Structure |
| fetchLMStructures | Fetch Structures |
| fetchLMSimulationAccounts | Fetch Accounts eligible for Simulation |
| fetchLMSimulation | Fetch Simulation Structures |
| readLMSimulation | View Simulation Structure details |
| createLMSimulation | Create Simulation Structure |
| downloadLMSimulationPDFReport | Download Simulation Advice |

This completes the entire configuration needed for consuming OBLM APIs in OBDX through OBRH.

10.2 Verify System Configurations

Following script helps in listing the LM specific System Configurations:-

```
SELECT prop_id as PROPERTY_IN_DATABASE, NVL(SUBSTR(t.UI_definition,
INSTR(t.UI_definition, "title")+9, INSTR(t.UI_definition, ",", "-11), t.UI_definition) AS
TITLE_ON_SCREEN, t.prop_value FROM digx_fw_config_var_b t where prop_id like '%LM%'
and module = 'OTHERMODULE' and determinant_value = '*'; -- Please enter correct determinant
value
```

Ensure correct values are maintained against the above properties.

This maintenance can be done from the “System Configuration” admin screen or directly in DB schema.

10.3 Enumerations

Following LM related enumerations are used in OBDX. They are used to fetch the values on the OBDX UI.

- a. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMSweepStatus';*
- b. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMStructureTypes';*
- c. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMCurrencyHolidayRates';*
- d. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMHolidayTreatment';*
- e. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMBackwardTreatment';*
- f. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMInterestMethod';*
- g. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMRellocationMethod';*
- h. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMStructureStatus';*
- i. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMStructureHostApprovalStatus';*
- j. *select * from DIGX_FW_ENUM_REPRESENTATIONS where enum_fqn='getLMChargeCollectionStatus';*

10.4 Adapter Properties

Certain fields (Eg: Enumerations, Status etc) can have different values in OBDX as compared to OBLM.

The mapping of all such values between OBDX and OBLM can be found/maintained using the below script:-

```
select * from DIGX_FW_CONFIG_ADAPTER_PROP_B where host_id = 'OBLM';
```

10.5 Simulation IC Group maintenance

While creating Simulation from OBDX, we need to send IC Group values to OBLM as per the below matrix:

| Structure Type | Interest Method | Scenario | PROP_ID |
|----------------|-----------------|----------|---------|
|----------------|-----------------|----------|---------|

| | | | |
|--------|----------|--|------------------------------|
| Sweep | Interest | All Participating accounts | SIM_SWEEP_INTEREST_ACCOUNT |
| Hybrid | Interest | Notional Header | SIM_HYBRID_INTEREST_NOTIONAL |
| Pool | Interest | Notional Header | SIM_POOL_INTEREST_NOTIONAL |
| | Advance | Notional Header | SIM_POOL_ADVANCE_NOTIONAL |
| | | All Participating Accounts (Except Notional) | SIM_POOL_ADVANCE_ACCOUNT |
| | Ratio | All Participating Accounts (Except Notional) | SIM_POOL_RATIO_ACCOUNT |

The values of these IC Groups can be different in each environment based on the IC Groups created in the respective OBLM system.

Certain properties have been created in OBDX, from where the values of these IC Groups shall be dynamically picked and sent to OBLM during Simulation creation from OBDX.

Below are the sample scripts to update those property values:-

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_SWEEP_INTEREST_ACCOUNT>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_SWEEP_INTEREST_ACCOUNT';
```

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_HYBRID_INTEREST_NOTIONAL>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_HYBRID_INTEREST_NOTIONAL';
```

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_POOL_INTEREST_NOTIONAL>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_POOL_INTEREST_NOTIONAL';
```

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_POOL_ADVANCE_NOTIONAL>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_POOL_ADVANCE_NOTIONAL';
```

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_POOL_ADVANCE_ACCOUNT>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_POOL_ADVANCE_ACCOUNT';
```

```
UPDATE DIGX_FW_CONFIG_ADAPTER_PROP_B SET
PROP_VALUE='<SIM_POOL_RATIO_ACCOUNT>' WHERE HOST_ID='OBLM' AND
TRANSACTION_TYPE='INTEREST_CALCULATION_GROUP' AND
PROP_ID='SIM_POOL_RATIO_ACCOUNT';
```

[Home](#)

11. User Credential Configuration

For some of the Mid-Office Products (OBVAM, OBTFPM, OBSCF, OBCM, INV, ASP, OBCFPM, MO_IPM) by default user credential configuration is DB-Based. However, it should be changed to use connector based configuration.

Following are the steps to change user credential configuration from DB-Based to connector based

-

- i. Update CREDENTIAL_STORE_TYPE property in table DIGX_FW_CONFIG_OUT_RS_CFG_B to “credential_impl” for the particular service ID.

Sample Script-

```
UPDATE DIGX_FW_CONFIG_OUT_RS_CFG_B set  
CREDENTIAL_STORE_TYPE='credential_impl' where SERVICE_ID='tokenOBTFPM142';
```

- ii. Create/Update required Connector Credentials mapping in weblogic console for particular Host (Outbound Connection) by referring to **Oracle Banking Digital Experience Connector Credential Store Guide.pdf**

[Home](#)

12. OBRH Integration Configuration

To consume Mid-Office APIs in OBDX using OBRH, following configurations need to be completed:

1. To integrate OBRH with OBDX, first some generic configurations and scripts needs to be executed. The Details for the same can be referred from **Oracle Banking Digital Experience Host Integration Guide; Section 7: Configurations for OBRH Integration.**

Note: Please skip adding entries to call OBRH end-point from adapters for already provided out of the box integrations from OBDX.

2. For Consuming Mid-Office Product services via OBRH, where OBDX will act as a consumer for OBRH, OBDX Consumer configurations required by OBRH needs to be imported in OBRH. The File to be imported would be present at the following location:
/installables/obrh/OBDX_Consumer.json.

Refer **OBRH User Manual, Import Service Consumer** section for how to import a consumer JSON in OBRH

3. Once the import is done successfully, you need to update the mid-office service provider's default implementation for IP, Port, Token Username and Token Password. Refer **OBRH User Manual, Add/Edit Implementation** section for achieving the same.

Note:

- When using OBRH there is no specific host implementation adapters. We use the third party adapter implementation for all services. The request and response specifications sent and received from OBRH for an end-point can be referred from the following: **externalinterface-api.zip**

- Any other assistance required regarding OBRH, you could refer the **OBRH User Manual.**
 - Also if anymore custom fields need to be sent to host or more fields are need to be configured in response; the following changes needs to be done
 - a. Fields needs to be added in OBDX Request and Response
 - b. Transformations needs to be changed in OBRH. Refer **OBRH User Manual, Request and Response Transformation Section.**
-

[Home](#)