

Oracle® Banking Platform Collections

Adapter Configuration Guide

Release 2.3.0.0.0

E56276-01

July 2014

Copyright © 2011, 2014, Oracle and/or its affiliates. All rights reserved.

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.

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.

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

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

Contents

Preface	ix
Audience	ix
Documentation Accessibility	ix
Organization of the Guide	ix
Conventions	x
1 Introduction	
1.1 Adapter Architecture.....	1-1
1.2 Product-Shipped Adapters.....	1-1
1.3 Calling an Adapter	1-3
1.3.1 How to call an Adapter.....	1-3
1.4 Implementing a new Adapter.....	1-4
2 List of Adapter APIs	
2.1 getSQLExpression.....	2-1
2.2 executeRule	2-1
2.3 retrieveJointAccountHolders	2-2
2.4 retrieveJointAccountForParty	2-3
2.5 retrieveLinkedAccountInfoFromLoans	2-4
2.6 retrieveLinkedAccountInfoFromCasa	2-4
2.7 retrieveLinkedAccountInfoFromTD	2-5
2.8 fetchTDPrincipleBalance.....	2-6
2.9 updateInCollectionIndicator	2-6
2.10 manualAccountWriteOffProcess	2-7
2.11 perFormAccountFullWriteOff	2-8
2.12 invokePartyWarningIndicatorApplicationService	2-8
2.13 invokeContactPointApplicationService.....	2-9
2.14 addHardshipReliefRequest	2-10
2.15 updateHardshipReliefRequest.....	2-11
2.16 generateDocument.....	2-12
2.17 generateAndDeliverDocument.....	2-13
2.18 registerEventsForActivity	2-14
2.19 fetchProductGroupDetailsOnBusinessUnit	2-15
2.20 populateHelperDTO.....	2-15
2.21 paymentTransferROSO.....	2-16

2.22	save.....	2-18
2.23	fetchDataFromHostSystem.....	2-19
2.24	initiateRosoProcessForApprovalProcess.....	2-19
2.25	getFactDetailsForRule	2-20
2.26	executeRuleWithPrePopulatedFactDetails	2-21
2.27	manualAccountWriteOffProcess	2-22
2.28	updateCollateralRealizationStatus	2-22
2.29	abortCompositeInstancesByTitle	2-23
2.30	sendMail	2-24
2.31	getFinancialDates	2-24
2.32	fetchTermDepositDetails	2-25
2.33	fetchTDAccount	2-26
2.34	initiateHardshipForApprovalProcess.....	2-26
2.35	updateIndividualDemographics	2-27
2.36	isWorkingDay.....	2-28
2.37	deleteWorklistItemForApplicationId.....	2-29

List of Tables

1-1	Product-Shipped Adapters.....	1-2
2-1	getSQLExpression Input Parameter.....	2-1
2-2	getSQLExpression Output Type.....	2-1
2-3	getSQLExpression OBP Service Classes.....	2-1
2-4	executeRule Input Parameter.....	2-2
2-5	executeRule Output Type.....	2-2
2-6	executeRule OBP Service Classes.....	2-2
2-7	retrieveJointAccountHolders Input Parameter.....	2-2
2-8	retrieveJointAccountHolders Output Type.....	2-3
2-9	retrieveJointAccountHolders OBP Service Classes.....	2-3
2-10	retrieveJointAccountForParty Input Parameter.....	2-3
2-11	retrieveJointAccountForParty Output Type.....	2-3
2-12	retrieveJointAccountForParty OBP Service Classes.....	2-3
2-13	retrieveLinkedAccountInfoFromLoans Input Parameter.....	2-4
2-14	retrieveLinkedAccountInfoFromLoans Output Type.....	2-4
2-15	retrieveLinkedAccountInfoFromLoans OBP Service Classes.....	2-4
2-16	retrieveLinkedAccountInfoFromCasa Input Parameter.....	2-4
2-17	retrieveLinkedAccountInfoFromCasa Output Type.....	2-5
2-18	retrieveLinkedAccountInfoFromCasa OBP Service Classes.....	2-5
2-19	retrieveLinkedAccountInfoFromTD Input Parameter.....	2-5
2-20	retrieveLinkedAccountInfoFromTD Output Type.....	2-5
2-21	retrieveLinkedAccountInfoFromTD OBP Service Classes.....	2-5
2-22	fetchTDPrincipleBalance Input Parameter.....	2-6
2-23	fetchTDPrincipleBalance Output Type.....	2-6
2-24	fetchTDPrincipleBalance OBP Service Classes.....	2-6
2-25	updateInCollectionIndicator Input Parameter.....	2-6
2-26	updateInCollectionIndicator Output Type.....	2-7
2-27	updateInCollectionIndicator OBP Service Classes.....	2-7
2-28	manualAccountWriteOffProcess Input Parameter.....	2-7
2-29	manualAccountWriteOffProcess Output Type.....	2-7
2-30	manualAccountWriteOffProcess OBP Service Classes.....	2-7
2-31	perFormAccountFullWriteOff Input Parameter.....	2-8
2-32	perFormAccountFullWriteOff Output Type.....	2-8
2-33	perFormAccountFullWriteOff OBP Service Classes.....	2-8
2-34	invokePartyWarningIndicatorApplicationService Input Parameter.....	2-9
2-35	invokePartyWarningIndicatorApplicationService Output Type.....	2-9
2-36	invokePartyWarningIndicatorApplicationService OBP Service Classes.....	2-9
2-37	invokeContactPointApplicationService Input Parameter.....	2-9
2-38	invokeContactPointApplicationService Output Type.....	2-10
2-39	invokeContactPointApplicationService OBP Service Classes.....	2-10
2-40	addHardshipReliefRequest Input Parameter.....	2-10
2-41	addHardshipReliefRequest Output Type.....	2-11
2-42	addHardshipReliefRequest OBP Service Classes.....	2-11
2-43	updateHardshipReliefRequest Input Parameter.....	2-11
2-44	updateHardshipReliefRequest Output Type.....	2-11
2-45	updateHardshipReliefRequest OBP Service Classes.....	2-12
2-46	generateDocument Input Parameter.....	2-12
2-47	generateDocument Output Type.....	2-12
2-48	generateDocument OBP Service Classes.....	2-13
2-49	generateAndDeliverDocument Input Parameter.....	2-13
2-50	generateAndDeliverDocument Output Type.....	2-14
2-51	generateAndDeliverDocument OBP Service Classes.....	2-14
2-52	registerEventsForActivity Input Parameter.....	2-14

2-53	registerEventsForActivity Output Type.....	2-14
2-54	registerEventsForActivity OBP Service Classes	2-14
2-55	fetchProductGroupDetailsOnBusinessUnit Input Parameter	2-15
2-56	fetchProductGroupDetailsOnBusinessUnit Output Type	2-15
2-57	fetchProductGroupDetailsOnBusinessUnit OBP Service Classes	2-15
2-58	populateHelperDTO Input Parameter.....	2-16
2-59	populateHelperDTO Output Type.....	2-16
2-60	populateHelperDTO OBP Service Classes	2-16
2-61	paymentTransfer Input Parameter	2-16
2-62	paymentTransfer Output Type	2-17
2-63	paymentTransfer OBP Service Classes	2-17
2-64	save Input Parameter.....	2-18
2-65	save Output Type.....	2-18
2-66	save OBP Service Classes	2-18
2-67	fetchDataFromHostSystem Input Parameter.....	2-19
2-68	fetchDataFromHostSystem Output Type.....	2-19
2-69	fetchDataFromHostSystem OBP Service Classes	2-19
2-70	initiateProcessForApproval Input Parameter.....	2-20
2-71	initiateProcessForApproval Output Type.....	2-20
2-72	initiateProcessForApproval OBP Service Classes	2-20
2-73	getFactDetailsForRule Input Parameter	2-20
2-74	getFactDetailsForRule Output Type	2-21
2-75	getFactDetailsForRule OBP Service Classes.....	2-21
2-76	executeRuleWithPrePopulatedFactDetails Input Parameter	2-21
2-77	executeRuleWithPrePopulatedFactDetails Output Type	2-21
2-78	executeRuleWithPrePopulatedFactDetails OBP Service Classes.....	2-22
2-79	manualAccountWriteOffProcess Input Parameter	2-22
2-80	manualAccountWriteOffProcess Output Type	2-22
2-81	manualAccountWriteOffProcess OBP Service Classes	2-22
2-82	updateCollateralRealizationStatus Input Parameter	2-23
2-83	updateCollateralRealizationStatus Output Type	2-23
2-84	updateCollateralRealizationStatus OBP Service Classes	2-23
2-85	abortCompositeInstancesByTitle Input Parameter	2-23
2-86	abortCompositeInstancesByTitle Output Type	2-23
2-87	abortCompositeInstancesByTitle OBP Service Classes	2-24
2-88	sendMail Input Parameter	2-24
2-89	sendMail Output Type	2-24
2-90	sendMail OBP Service Classes	2-24
2-91	getFinancialDates Input Parameter.....	2-25
2-92	getFinancialDates Output Type.....	2-25
2-93	getFinancialDates OBP Service Classes	2-25
2-94	fetchTermDepositDetails Input Parameter	2-25
2-95	fetchTermDepositDetails Output Type	2-25
2-96	fetchTermDepositDetails OBP Service Classes.....	2-26
2-97	fetchTDAccount Input Parameter	2-26
2-98	fetchTDAccount Output Type	2-26
2-99	fetchTDAccount OBP Service Classes.....	2-26
2-100	initiateHardshipForApprovalProcess Input Parameter.....	2-27
2-101	initiateHardshipForApprovalProcess Output Type.....	2-27
2-102	initiateHardshipForApprovalProcess OBP Service Classes	2-27
2-103	updateIndividualDemographics Input Parameter	2-27
2-104	updateIndividualDemographics Output Type	2-28
2-105	updateIndividualDemographics OBP Service Classes.....	2-28
2-106	isWorkingDay Input Parameter.....	2-28
2-107	isWorkingDay Output Type.....	2-28

2-108	isWorkingDay OBP Service Classes.....	2-28
2-109	deleteWorklistItemForApplicationId Input Parameter.....	2-29
2-110	deleteWorklistItemForApplicationId Output Type.....	2-29
2-111	deleteWorklistItemForApplicationId OBP Service Classes.....	2-29

Preface

This guide helps you to understand the Collection adapters and its associated operations.

This preface contains the following topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Organization of the Guide](#)
- [Conventions](#)

Audience

This document is intended for the following:

- IT Deployment Team
- Consulting Staff
- Administrators

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 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.

Organization of the Guide

Chapter 1, "Introduction"

This chapter presents an overview of adapters and various product-shipped adapters.

Chapter 2, "List of Adapter APIs"

This chapter provides detailed information on adapter APIs.

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.
<i>italic</i>	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.

Introduction

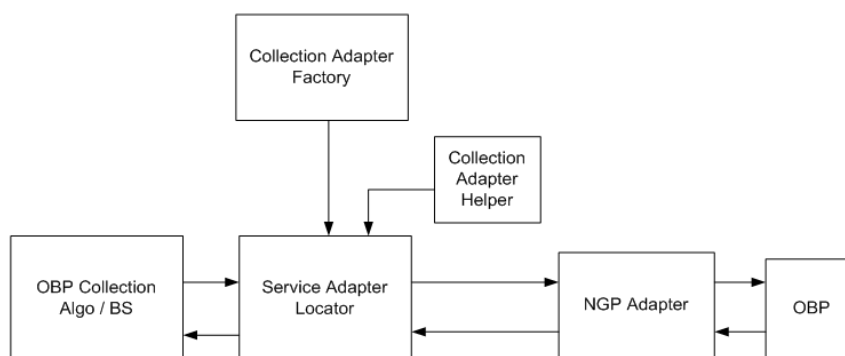
Oracle Banking Platform Collections module interacts with multiple systems available within the enterprise arena. Transaction data owned by banking host systems are forwarded to OBP Collections application. Updates on the transaction data in the Collections module are updated on the banking host and other integrated modules/systems as well. In addition, Collection application is designed to integrate with any other services by using the adapter pattern. Collections provide adapters to integrate with OBP modules and technical modules such as Document Management System (DMS)/Alert/E-mail systems.

Adapter pattern allows consulting or implementation team to integrate Collection application with any other host or enterprise service. This document covers the details of the adapters being shipped with Collections module. Additionally, it also covers the steps to create any new adapter.

1.1 Adapter Architecture

The Adapter Architecture Flow diagram is as follows:

Figure 1–1 Adapter Architecture



1.2 Product-Shipped Adapters

Adapter properties must be present in `flx_fw_config_all_b` table under category id '`CollectionAdapterFactory`'. Factory returns the implementation class based on `flx_fw_config_all_b.prop_value` returned from table. OBP Collections provide different set of implementation classes to call OBP services. These implementation classes

invoke OBP services to get the data into the DTO. OBP module DTO's are different from the DTO used in Collections. The helper classes are used to convert the data received or sent to OBP services into Collections specific DTO. This design pattern helps in creating an abstraction layer between the OBP modules and Collections module.

The following table shows the list of adapter classes created in Collections along with their corresponding helper classes. These Collections implementation classes internally invoke services from respective HOST. For example, if the adapter is for OBP, then these classes will invoke services from OBP.

Table 1–1 Product-Shipped Adapters

Adapter	Implementation Class	Description
NGP_HOST_IMPL	com.splwg.ccb.domain.externalservice.adapter.impl.CollectionNGPAdapter	OBP implementation class. Use this adapter to interact with OBP.
NGP_HOST_HELPER	com.splwg.ccb.domain.externalservice.adapter.helper.NGPCollectionAdapterHelper	This is a helper class that converts the Collection DTO object to OBP host DTO object.
DOC_HOST_IMPL	com.splwg.ccb.domain.externalservice.adapter.impl.CollectionNGPAdapter	This is an implementation class, which generates the documents. If you want to use some other interface to generate document, then you must use a separate implementation class.
DOC_HOST_HELPER	com.splwg.ccb.domain.externalservice.adapter.helper.NGPCollectionAdapterHelper	This helper class creates a DTO, which is required to generate the document.
EVENT_HOST_IMPL	com.splwg.ccb.domain.externalservice.adapter.impl.CollectionNGPAdapter	This adapter is used to send e-mails and SMS.
EVENT_HOST_HELPER	com.splwg.ccb.domain.externalservice.adapter.helper.NGPCollectionAdapterHelper	This adapter converts the DTO of one module to another module. This helper class creates DTO required for event management API.
APPROVAL_HOST_IMPL	com.splwg.ccb.domain.externalservice.adapter.impl.CollectionNGPAdapter	This adapter is used for approval workflow.
APPROVAL_HOST_HELPER	com.splwg.ccb.domain.externalservice.adapter.helper.NGPCollectionAdapterHelper	This adapter converts the DTO of one module to another module. This helper class creates DTO for approval workflow API.

The naming conventions for the classes in the adapter property stored in the table **flx_fw_config_all_b.prop_id** are:

Implementation class for the host.

[Name of the Host]_HOST_HOST_IMPL

This class invokes the services exposed by host

Helper class for the host

[Name of the Host]_HOST_HOST_HELPER

This class converts the DTO of one module to another module.

1.3 Calling an Adapter

Adapters are invoked when modules need to interact with each other or with any other application. Adapters are invoked by the following:

- **Transactions:** When there is data updation in host, such as collateral updates, warning indicators update, collection flag update in host, and so on.
- **Infrastructure:** When Infrastructure services are called by ORMB, such as Rule engine, DMS, Approval system, and so on.

1.3.1 How to call an Adapter

The following information explains how to call an adapter:

- **Transaction**

The transaction updates the collateral's realization status in the OBP host. Here, the client class passes **NULL** as first parameter and **host id** as second parameter. The Collection adapter factory then adds **_HOST_IMPL** to retrieve the key value.

```
CollectionAdapterFactory adaptorFactory =
CollectionAdapterFactory.getInstance();
```

```
ngpAdaptor = adaptorFactory ICollectionAdapter.getAdapter(null, hostId);
```

Based on above configuration, the hosted factory returns the host implementation class. Currently, we support OBP as host.

- **Infrastructure**

Here, the client class needs to pass the **INFRA component id** as first parameter and **NULL** as second parameter. The Collection adapter factory class then adds **_HOST_IMPL** to retrieve the key value.

These implementations are maintained separately for different usage.

RULE

```
CollectionAdapterFactory adaptorFactory =
CollectionAdapterFactory.getInstance();
```

```
ICollectionAdapter ngpAdaptor = adaptorFactory.getAdapter("RULE", null);
```

DMS

```
CollectionAdapterFactory adaptorFactory =
CollectionAdapterFactory.getInstance();
```

```
ICollectionAdapter ngpAdaptor = adaptorFactory.getAdapter("DOC", null);
```

EVENT(SMS/EMAIL)

```
CollectionAdapterFactory adaptorFactory =
CollectionAdapterFactory.getInstance();
```

```
ICollectionAdapter ngpAdaptor = adaptorFactory.getAdapter("EVENT", null);
```

APPROVAL

```
CollectionAdapterFactory adaptorFactory =
CollectionAdapterFactory.getInstance();
```

```
ICollectionAdapter ngpAdaptor = adaptorFactory.getAdapter("APPROVAL",
null);
```

1.4 Implementing a new Adapter

To implement a new adapter, follow the steps:

1. Create a new row in the flx_fw_config_all_b having values:
 - PROP_ID = [Name of the Host]_HOST_IMPL
 - CATEGORY_ID = 'CollectionAdapterFactory'
 - PROP_VALUE = new implementation class which should implement the ICollectionAdapter class
 - FACTORY_SHIPPED_FLAG = 'N'
2. Write a new implementation class for implementation host.
3. Create a new row in the flx_fw_config_all_b having values:
 - PROP_ID = [[Name of the Host]_HOST_HELPER
 - CATEGORY_ID = 'CollectionAdapterFactory'
 - PROP_VALUE = new implementation class which should implement the ICollectionAdapterHelper class
 - FACTORY_SHIPPED_FLAG = 'N'
4. Write a new implementation class for the helper host.
5. Put appropriate summary, comments, version and date for fields PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY.
6. Put OBJECT_STATUS_FLAG = 'A' and OBJECT_VERSION_NUMBER = '1'.

The newly created implementation class may not require implementing all methods available in the ICollectionAdapter class. Hence, the methods that are not required to implement can be left blank. Similarly, the helper class implementation unused methods can have blank implementation.

Reference query to add new adapter entry:

```
INSERT INTO flx_fw_config_all_b
(PROP_ID, CATEGORY_ID, PROP_VALUE, FACTORY_SHIPPED_FLAG, PROP_COMMENTS,
SUMMARY_TEXT, CREATED_BY, CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_
DATE, OBJECT_STATUS_FLAG, OBJECT_VERSION_NUMBER)
VALUES
('RULE_HOST_IMPL', 'CollectionAdapterFactory',
'com.splwg.ccb.domain.externalservice.adapter.impl.CollectionNGPAdapter',
'y', 'Collection adapter data', 'Collection adapter data', 'ofssuser',
SYSDATE, 'ofssuser', SYSDATE, 'A', 1);
```

List of Adapter APIs

Following Adapter APIs are available in OBP Collections:

2.1 getSQLExpression

Description:

This method is used to fetch the condition created by using **Filter Author** OBP screen.

Syntax:

```
public String getSQLExpression(String filterId)
```

Input:

Table 2–1 *getSQLExpression Input Parameter*

Parameter	Type	Description	Mandatory
filterId	String	This is the name of the filter configured in OBP screen, Back Office > Rules > Filter Definition.	Y

Output:

Table 2–2 *getSQLExpression Output Type*

Type	Description
String	Returns the SQL expression after where clause.

OBP Service Name:

Table 2–3 *getSQLExpression OBP Service Classes*

NGP Adapter Class Name	CollectionRulesAdapter
Method name	getSQLExpression(String filterId)
OBP Application Service Name	FilterDerivationApplicationService
Method name	deriveSQLFilter(sessionContext, filterId)

2.2 executeRule

Description:

This method is used to call the rule engine to process rules created in OBP.

Syntax:

```
public Map<String, RuleFactInfoDto> executeRule(String ruleId, String
ruleEffectiveDate, List<RuleFactInfoDto> ruleFactInfoList, boolean isRuleSet)
```

Input:**Table 2–4** *executeRule Input Parameter*

Parameter	Type	Description	Mandatory
ruleId	String	This is the name of the filter configured in OBP screen, Back Office->Rules->Rule Author.	Y
ruleEffectiveDate	String	Rule effective date.	N
ruleFactInfoList	List<RuleFactInfoDto>	List of facts to be used in rule	N
isRuleSet	boolean	Identifier for rule / rule set	Y

Output:**Table 2–5** *executeRule Output Type*

Type	Description
Map<String, RuleFactInfoDto>	Returns the rules outcome.

OBP Service Name:**Table 2–6** *executeRule OBP Service Classes*

OBP Adapter Class Name	CollectionRulesAdapter
Method name	executeRule(String ruleId, String ruleEffectiveDate, FactInformationDTO[] factInformationDTOs)
OBP Application Service Name	RuleExecutionApplicationService
Method name	executeLatestRuleForGivenDate(sessionContext, ruleContext,null)

2.3 retrieveJointAccountHolders

Description:

This method is used to retrieve the list of account holders for an account.

Syntax:

```
public List<String> retrieveJointAccountHolders(String accountId)
```

Input:**Table 2–7** *retrieveJointAccountHolders Input Parameter*

Parameter	Type	Description	Mandatory
accountId	String	Indicates the host account number.	Y

Output:**Table 2–8** *retrieveJointAccountHolders Output Type*

Type	Description
List<String>	Returns the list of all account holders for the Account ID entered.

OBP Service Name:**Table 2–9** *retrieveJointAccountHolders OBP Service Classes*

NGP Adapter Class Name	PartyAccountRelationshipTypeAdapter
Method Name	fetchAllPartiesOfAccount(accountId)
OBP Application Service Name	PartyAccountRelationshipApplicationService
Method name	fetchAllPartiesOfAccount(fetchSessionContext(), accountId)

2.4 retrieveJointAccountForParty

Description:

This method is used to retrieve joint account holders for a party.

Syntax:

```
public List<NgpAccountInformationBean> retrieveJointAccountForParty(String partyId)
```

Input:**Table 2–10** *retrieveJointAccountForParty Input Parameter*

Parameter	Type	Description	Mandatory
partyId	String	Indicates the host customer number.	Y

Output:**Table 2–11** *retrieveJointAccountForParty Output Type*

Type	Description
List<NgpAccountInformationBean>	Returns the list of all Account Information Bean, where the input partyId is joint account holder.

OBP Service Name:**Table 2–12** *retrieveJointAccountForParty OBP Service Classes*

NGP Adapter Class Name	PartyAccountRelationshipTypeAdapter
Method Name	fetchAllAccountsOfParty(partyId)
OBP Application Service Name	PartyAccountRelationshipApplicationService
Method name	fetchAllAccountsofParty(fetchSessionContext(), partyId)

2.5 retriveLinkedAccountInfoFromLoans

Description:

This method is used to retrieve OBP linked account information for a loan account type.

Syntax:

```
public NgpAccountInformationBean retriveLinkedAccountInfoFromLoans(String
accountId, Date systemDate)
```

Input:

Table 2–13 *retriveLinkedAccountInfoFromLoans Input Parameter*

Parameter	Type	Description	Mandatory
accountId	String	Indicates the host account number.	Y
systemDate	com.splwg.base.api.datatypes.Date	Indicates the system date.	Y

Output:

Table 2–14 *retriveLinkedAccountInfoFromLoans Output Type*

Type	Description
NgpAccountInformationBean	Retrieves linked account from account whose account type is LOAN.

OBP Service Name:

Table 2–15 *retriveLinkedAccountInfoFromLoans OBP Service Classes*

NGP Adapter Class Name	LinkedAccountInfoFromLoansAdapter
Method Name	inquireAccountInformation(accountId)
OBP Application Service Name	LoanAccountCollectionInformationApplicationService()
Method name	inquireAccountInformation(fetchSessionContext(), accountId)

2.6 retriveLinkedAccountInfoFromCasa

Description:

This method is used to retrieve OBP linked account information whose account type is CASA.

Syntax:

```
public NgpAccountInformationBean retriveLinkedAccountInfoFromCasa(String
accountId)
```

Input:

Table 2–16 *retriveLinkedAccountInfoFromCasa Input Parameter*

Parameter	Type	Description
accountId	String	Indicates the host account number.

Output:**Table 2–17 retrieveLinkedAccountInfoFromCasa Output Type**

Type	Description
NgpAccountInformationBean	Retrieves linked account from account whose account type is CASA.

OBP Service Name:**Table 2–18 retrieveLinkedAccountInfoFromCasa OBP Service Classes**

NGP Adapter Class Name	LinkedAccountInfoFromCasaAdapter
Method Name	fetchAccountDetails(accountId)
OBP Application Service Name	DDAInquiryApplicationService
Method name	fetchAccountDetails(fetchSessionContext(), accountId)

2.7 retrieveLinkedAccountInfoFromTD

Description:

This method is used to retrieve OBP linked account Information whose account type is TD.

Syntax:

```
public ArrayList<NgpAccountInformationBean>
retriveLinkedAccountInfoFromTD(String partyId)
```

Input:**Table 2–19 retrieveLinkedAccountInfoFromTD Input Parameter**

Parameter	Type	Description	Mandatory
partyId	String	Indicates the host customer number.	Y

Output:**Table 2–20 retrieveLinkedAccountInfoFromTD Output Type**

Type	Description
NgpAccountInformationBean	Retrieves linked account from account whose account Type is TD.

OBP Service Name:**Table 2–21 retrieveLinkedAccountInfoFromTD OBP Service Classes**

NGP Adapter Class Name	LinkedAccountInfoFromTDAdapter
Method Name	fetchAllAccountsForPartyId(partyId)
OBP Application Service Name	TermDepositAccountApplicationService
Method name	fetchAllAccountsForPartyId(fetchSessionContext(), partyId)

2.8 fetchTDPrincipleBalance

Description:

This method is used to fetch the principle balance whose account type is TD.

Syntax:

```
public BigDecimal fetchTDPrincipleBalance(String accountId)
```

Input:

Table 2–22 *fetchTDPrincipleBalance Input Parameter*

Parameter	Type	Description	Mandatory
accountId	String	Indicates the host customer number.	Y

Output:

Table 2–23 *fetchTDPrincipleBalance Output Type*

Type	Description
BigDecimal	Returns the Accounts TD principle balance.

OBP Service Name:

Table 2–24 *fetchTDPrincipleBalance OBP Service Classes*

NGP Adapter Class Name	LinkedAccountBalanceFromTDAdapter
Method Name	fetchAllDeposits(accountId,null)
OBP Application Service Name	DepositApplicationService
Method name	fetchAllDeposits(fetchSessionContext(), accountId, depositStatus)

2.9 updateInCollectionIndicator

Description:

This method is used to update the **InCollection** indicator for account extension.

Syntax:

```
public boolean updateInCollectionIndicator(AccountExtension acctExtn)
```

Input:

Table 2–25 *updateInCollectionIndicator Input Parameter*

Parameter	Type	Description	Mandatory
acctExtn	com.splwg.ccb.domain.collection.accountExtension.AccountExtension	Indicates the collection account object.	Y

Output:**Table 2–26** *updateInCollectionIndicator Output Type*

Type	Description
boolean	Returns true if update happens successfully.

OBP Service Name:**Table 2–27** *updateInCollectionIndicator OBP Service Classes*

NGP Adapter Class Name	CollectionAcctInCollectionAdapter
Method Name	updateInCollectionIndicator(RecoveryAccountDetailsDTO) updateInCollectionIndicator(acctExtn.getAccountNumber())
OBP Application Service Name	DDAWarningIndicatorApplicationService
Method name	setRecoveryFlagWarningIndicator(sessionContext, recoveryAccountDetailsDTO)

2.10 manualAccountWriteOffProcess

Description:

This method is used to perform full account write-off for manual write-off type. The method sets the write-off request date as business date for the accounts in Collections. The manual write-off involves approval and rejection mechanism by BPEL process.

Syntax:

```
public TransactionStatus manualAccountWriteOffProcess(String accountNumber)
```

Input:**Table 2–28** *manualAccountWriteOffProcess Input Parameter*

Parameter	Type	Description	Mandatory
accountNumber	String	Account Number as stored in the host system.(Host Account Number)	Y

Output:**Table 2–29** *manualAccountWriteOffProcess Output Type*

Type	Description
TransactionStatus	Returns the transactionStatus object.

OBP Service Name:**Table 2–30** *manualAccountWriteOffProcess OBP Service Classes*

NGP Adapter Class Name	AccountWriteOffProcessCallingAdapter
Method Name	callAccountWriteOffProcess(accountNumber)
OBP Application Service Name	AccountWriteOffProcessApplicationService
Method name	callAccountWriteOffProcess(fetchSessionContext(), accountNumber)

2.11 perFormAccountFullWriteOff

Description:

This method is used to perform full account write-off for STP write-off type. The method sets the write-off request date as business date for the accounts in Collections.

Syntax:

```
public String perFormAccountFullWriteOff(String accountId, String accountType,
String reason1, String reason2)
```

Input:

Table 2–31 *perFormAccountFullWriteOff Input Parameter*

Parameter	Type	Description
accounted	String	Indicates the account number as stored in the host system.
accountType	String	Indicates the account type (LN / TD / CS).
reason1	String	Indicates the primary write-off reason.
reason2	String	Indicates the secondary write-off reason.

Output:

Table 2–32 *perFormAccountFullWriteOff Output Type*

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:

Table 2–33 *perFormAccountFullWriteOff OBP Service Classes*

NGP Adapter Class Name	CollectionAcctWriteOffAdapter
Method Name	perFormAccountFullWriteOff(accountId,accountType2, reason1, reason2)
OBP Application Service Name	AccountWriteOffApplicationService
Method name	performAccountFullWriteOff(sessionContext,accountId, accountType, reason1, reason2)

2.12 invokePartyWarningIndicatorApplicationService

Description:

This method is used to invoke **PartyWarning** indicator application service.

Syntax:

```
public String invokePartyWarningIndicatorApplicationService(String partyId, String
partyType,
String warningIndicatorType, boolean warningIndicatorValue)
```

Input:**Table 2–34** *invokePartyWarningIndicatorApplicationService Input Parameter*

Parameter	Type	Description
partyId	String	Indicates the host customer number.
partyType	String	Indicates the party type that is, IND (Individual), LEG (Legal), BAN (Bank), TRU (Trustee).
warningIndicator Type	String	Indicates the warning indicator type that is, DEC (Deceased).
warningIndicator Value	Boolean	Warning Indicator Boolean Value

Output:**Table 2–35** *invokePartyWarningIndicatorApplicationService Output Type*

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:**Table 2–36** *invokePartyWarningIndicatorApplicationService OBP Service Classes*

NGP Adapter Class Name	PartyWarningIndicatorAdapter
Method Name	publishPartyWarningIndicator(partyWarningIndicatorDTO)
OBP Application Service Name	PartyWarningIndicatorApplicationService
Method name	publishPartyWarningIndicator(fetchSessionContext(), partyWarningIndicatorDTO)

2.13 invokeContactPointApplicationService

Description:

This method is used to invoke contact point application service.

Syntax:

```
public String invokeContactPointApplicationService(String partyId, String
ContactPreferenceType, String contactPointType, boolean
marketingConsentFlagValue, String determinantVal, Date marketingConsentStartDt,
Date marketingConsentEndDt, String purposeType)
```

Input:**Table 2–37** *invokeContactPointApplicationService Input Parameter*

Parameter	Type	Description
partyId	String	Indicates the host customer number.
contactPreferenceType	String	Indicates the contact preference type.
contactPointType	String	Indicates the contact point type.
marketingConsentFlagValue	Boolean	Indicates the marketing consent value.
determinantVal	String	Indicates the determinant value.

Table 2–37 (Cont.) invokeContactPointApplicationService Input Parameter

Parameter	Type	Description
marketingConsentStartDt	com.splwg.base.api.datatypes.Date	Indicates the marketing consent start date.
marketingConsentEndDt	com.splwg.base.api.datatypes.Date	Indicates the marketing consent end date.
purposeType	String	Indicates the purpose type.

Output:**Table 2–38 invokeContactPointApplicationService Output Type**

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:**Table 2–39 invokeContactPointApplicationService OBP Service Classes**

NGP Adapter Class Name	ContactPointAdapter
Method Name	updateContactPoint(contactPointDTO)
OBP Application Service Name	contactPointApplicationService
Method name	fetchContactPoint(fetchSessionContext(), contactPointDto)

2.14 addHardshipReliefRequest

Description:

This method is used to add a hardship relief request.

Syntax:

```
public String addHardshipReliefRequest(List acctId, String partyId, String applicationId, String initiatedBy, String reliefEffectiveDt, String reliefExpiryDt, String applicationDt, String startDt, String endDt, String expiryDt, String approvedDate, String approvedBy)
```

Input:**Table 2–40 addHardshipReliefRequest Input Parameter**

Parameter	Type	Description	Mandatory
partyId	String	Indicates the host customer number.	Y
applicationId	String	Indicates the application ID.	Y
initiatedBy	String	Indicates the individual who initiated hardship relief request.	Y
reliefEffectiveDt	String	Indicates the relief effective date.	Y
reliefExpiryDt	String	Indicates the relief expiry date.	Y
acctId	List	Indicates the account ID.	Y
applicationDt	String	Indicates the application date.	Y
startDt	String	Indicates the start date.	Y

Table 2–40 (Cont.) addHardshipReliefRequest Input Parameter

Parameter	Type	Description	Mandatory
endDt	String	Indicates the end date.	Y
expiryDt	String	Indicates the expiry date.	Y
approvedDate	String	Indicates the approved date.	Y
approvedBy	String	Indicates the approver name.	Y

Output:

Table 2–41 addHardshipReliefRequest Output Type

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:

Table 2–42 addHardshipReliefRequest OBP Service Classes

NGP Adapter Class Name	HardshipReliefRequestAdapter
Method Name	addHardshipReliefRequest(hardshipReliefRequestDTO)
OBP Application Service Name	HardshipReliefRequestApplicationService
Method name	addHardshipReliefRequest(fetchSessionContext(), hardshipReliefRequestDTO)

2.15 updateHardshipReliefRequest

Description:

This method is used to update the hardship relief request.

Syntax:

```
public String updateHardshipReliefRequest(String applicationId, String partyId,
String hardshipReliefUpdataionType, Date hardshipExtensionDate)
```

Input:

Table 2–43 updateHardshipReliefRequest Input Parameter

Parameter	Type	Description	Mandatory
partyId	String	Indicates the host customer number.	Y
applicationId	String	Indicates the application ID.	Y
hardshipReliefUpd ataionType	String	Indicates the relief update type.	Y
hardshipExtension Date	com.splwg.base.ap i.datatypes.Date	Indicates the hardship extension date.	Y

Output:

Table 2–44 updateHardshipReliefRequest Output Type

Type	Description
String	Returns null for success, else returns error code.

OBP Service Name:**Table 2–45 updateHardshipReliefRequest OBP Service Classes**

NGP Adapter Class Name	HardshipReliefRequestAdapter
Method Name	updateHardshipReliefRequest(applicationId, partyId,hardshipReliefUpdataionType, (hardshipExtensionDate!=null))
OBP Application Service Name	HardshipReliefRequestApplicationService
Method name	updateHardshipReliefRequest(fetchSessionContext(), applicationId,partyId,HardshipReliefUpdataionType.Cancellation,null)

2.16 generateDocument

Description:

This method is used to generate document. It takes input as **CollOutboundDocumentRequestDTO** for which the required parameters are Customer ID (Party ID in OBP host), Customer Contact ID and Template ID (Document Type in OBP host). The parameters specific to report can be specified in **CollReportRequestDTO** in form of key value pairs. Document is generated for given Document Type using configured parameters and is stored on server.

Also, the search indexes are created for PARTY_ID and CUSTOMER_CONTACT_ID. **CollDocumentGenerationResponse** is response to document generation process, which contains generated document id.

Syntax:

```
Public CollDocumentGenerationResponse
generateDocument(CollOutboundDocumentRequestDTO
outboundDocumentRequestDTO)
```

Input:**Table 2–46 generateDocument Input Parameter**

Parameter	Type	Description	Mandatory
outboundDocumentRequestDTO	CollOutboundDocumentRequestDTO	Request object for document generation. Mandatory fields are HOST CUSTOMER ID (PARTY ID in OBP host), CUSTOMER CONTACT ID TEMPLATE ID (DOCUMENT TYPE in OBP host)	Y

Output:**Table 2–47 generateDocument Output Type**

Type	Description
CollDocumentGenerationResponse	It contains generated Document ID.

OBP Service Name:**Table 2–48 generateDocument OBP Service Classes**

NGP Adapter Class Name	CollectionDMSAdapter
Method Name	generateDocument(OutboundDocumentRequestDTO outboundDocumentRequestDTO)
OBP Application Service Name	OutboundDocumentApplicationService
Method name	generateDocument(SessionContext sessionContext,OutboundDocumentRequestDTO outboundDocumentRequestDTO)

2.17 generateAndDeliverDocument

Description:

This method is responsible to generate and deliver document to the respective customer.

It takes input as CollOutboundDocumentRequestDTO, for which the required parameters are Customer ID (Party ID in OBP host), Customer Contact ID and Template ID (Document Type in OBP host). Parameters specific to report can be specified in the CollReportRequestDTO in form of key-value pairs, which are optional. Document is generated for given Document Type using configured Parameters and is stored on server.

Also, search indexes are created for PARTY_ID and CUSTOMER_CONTACT_ID. Further, contact preferences and delivery channel are resolved based on given Customer ID. Document is delivered to respective Customer.

CollOutboundDocumentResponse is returned as response, which holds CollDocumentGenerationResponse and CollDocumentDeliveryResponse.

CollDocumentGenerationResponse is response to document generation process, which contains generated document id.

Syntax:

```
public CollOutboundDocumentResponse
generateAndDeliverDocument(CollOutboundDocumentRequestDTO
collOutboundDocumentRequestDTO)
```

Input:**Table 2–49 generateAndDeliverDocument Input Parameter**

Parameter	Type	Description	Mandatory
collOutboundDocumentRequestDTO	CollOutboundDocumentRequestDTO	Request object for document generation. Mandatory fields are HOST CUSTOMER ID (PARTY ID in OBP host), CUSTOMER CONTACT ID TEMPLATE ID (DOCUMENT TYPE in OBP host)	Y

Output:**Table 2–50 generateAndDeliverDocument Output Type**

Type	Description
CollDocumentGenerationResponse	Response objects for document generation and delivery. It contains generated document Id.

OBP Service Name:**Table 2–51 generateAndDeliverDocument OBP Service Classes**

NGP Adapter Class Name	CollectionDMSAdapter
Method Name	generateAndDeliverDocument(OutboundDocumentRequestDTO outboundDocumentRequestDTO)
OBP Application Service Name 1	OutboundDocumentApplicationService
Method name	generateDocument(SessionContext sessionContext, OutboundDocumentRequestDTO outboundDocumentRequestDTO)
OBP Application Service Name 2	OutboundDocumentDeliveryApplicationService
Method name	deliverDocumentForParty(SessionContext sessionContext, DocumentDTO documentDTO)

2.18 registerEventsForActivity

Description:

This method is used to register events for activities.

Syntax:

```
public String registerEventsForActivity(AlertDTO alertDTO) throws Exception
```

Inputs:**Table 2–52 registerEventsForActivity Input Parameter**

Parameter	Type	Description	Mandatory
alertDTO	com.splwg.ccb.domain.externalser vice.adapter.dto.alert.AlertDTO	Indicates the Alert DTO.	Y

Output:**Table 2–53 registerEventsForActivity Output Type**

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:**Table 2–54 registerEventsForActivity OBP Service Classes**

NGP Adapter Class Name	CollectionAlertAdapter
Method Name	registerEventsForActivity(alertDTO.getActivityId(), collectionAlertDTO)
OBP Application Service Name	ActivityRegistrationApplicationService

Table 2–54 (Cont.) registerEventsForActivity OBP Service Classes

Method name	registerActivityAndGenerateConfiguredEvents(fetchSessionContext(), activityId, collectionActivityLogDTO)
-------------	----------------------------------------------------------------------------------------------------------

2.19 fetchProductGroupDetailsOnBusinessUnit

Description:

This method is used to fetch product group details.

Syntax:

```
public ReliefTypesHelperDTO fetchProductGroupDetailsOnBusinessUnit(String productGroupCode, String businessUnit)
```

Input:

Table 2–55 fetchProductGroupDetailsOnBusinessUnit Input Parameter

Parameter	Type	Description	Mandatory
productGroupCode	String	Indicates the product group code.	Y
businessUnit	String	Indicates the business unit.	Y

Output:

Table 2–56 fetchProductGroupDetailsOnBusinessUnit Output Type

Type	Description
ReliefTypesHelperDTO	Returns ReliefTypesHelperDTO object.

OBP Service Name:

Table 2–57 fetchProductGroupDetailsOnBusinessUnit OBP Service Classes

NGP Adapter Class Name	ProductManufacturingAdapter
Method Name	fetchProductGroupDetailsOnBusinessUnit(productGroupCode,businessUnit)
OBP Application Service Name	ProductManufacturingApplicationService
Method name	fetchProductGroupDetailsOnBusinessUnit(fetchSessionContext(), productGroupCode, businessUnit)

2.20 populateHelperDTO

Description:

This method is used to populate the helper DTO.

Syntax:

```
private ReliefTypesHelperDTO populateHelperDTO(CreditPolicyTemplateDTO[] creditPolicyDTO)
```

Input:**Table 2–58** *populateHelperDTO Input Parameter*

Parameter	Type	Description	Mandatory
creditPolicyDTO	com.ofss.fc.app.pm.dto.creditpolicytemplate.CreditPolicyTemplateDTO	Indicates the credit policy.	Y

Output:**Table 2–59** *populateHelperDTO Output Type*

Type	Description
ReliefTypesHelperDTO	Returns ReliefTypesHelperDTO object.

OBP Service Name:**Table 2–60** *populateHelperDTO OBP Service Classes*

NGP Adapter Class Name
Method Name
OBP Application Service Name
Method name

2.21 paymentTransferROSO

Description:

This method is used to perform payment transfer for Right of Set-off (ROSO) process.

Payments performed are:

- Term Loan to Demand Deposit Account
- Term Loan to Term Deposit Account - It is a two step process,
 - First, the funds are transferred from Term Deposit to Ledger account and then from Ledger account to Term Loan account.
- Overdraft to Demand Deposit Account
- Overdraft to Term Deposit Account

Syntax:

```
public Map<String, String> paymentTransferROSO(String delinquentAcctId, String
hostProductClassCd, String targetAccountnumber, String amountToDeduct, String
targetProductCode, String currencyCode, String processDate, String glAccount, String
depositId)
```

Input:**Table 2–61** *paymentTransfer Input Parameter*

Parameter	Type	Description	Mandatory
delinquentAcctId	String	Delinquent (Term Loan/Overdraft) account in which funds needs to be credited.	Y

Table 2–61 (Cont.) paymentTransfer Input Parameter

Parameter	Type	Description	Mandatory
hostProductClassCd	String	Indicates the host product class code of delinquent account.	Y
targetAccountnumber	String	Indicates the target (Demand Deposit/Term Deposit) account number from which funds needs to be transferred.	Y
amountToDeduct	String	Indicates the amount of funds to be transferred.	Y
targetProductCode	String	Indicates the product code of target account.	Y
currencyCode	String	Currency in which funds are transferred.	Y
processDate	String	Date on which funds are transferred	Y
glAccount	String	Ledger account number. It is used when funds are transferred from Term Deposit to Term Loan account. Mandatory, when funds are transferred from Term Deposit to Term Loan account	N
depositId	String	Mandatory when funds are transferred from Term Deposit account.	N

Output:**Table 2–62 paymentTransfer Output Type**

Type	Description
Map<String, String>	Payment transaction response. It is a map of status code and transaction Id (key = status code, value= transaction Id). Status Code '0' represents successful transaction.

OBP Service Name:**Table 2–63 paymentTransfer OBP Service Classes**

NGP Adapter Class Name	PerformPaymentAdapter
Method Name 1	doInstallmentPayment(String accountId, TransactionDetailsDTO transactionDetailsDTO, int advancePaymentCounter, ComponentPreferenceDTO componentPreferenceDTO)
OBP Application Service Name	LoanInstallmentPaymentApplicationService
Method name	doInstallmentPayment(SessionContext sessionContext, String accountId, TransactionDetailsDTO transactionDetailsDTO, int advancePaymentCounter, ComponentPreferenceDTO componentPreferenceDTO)
Method Name 2	transferFundsToBeneficiaries(FundTransferTransactionDTO fundsTransferTransactionDTO)
OBP Application Service Name	DemandDepositFundsTransferService
Method name	transferFundsToBeneficiaries(SessionContext sessionContext ,FundTransferTransactionDTO undsTransferTransactionDTO)
Method Name 3	inquireAndDoRedemption(RedemptionRequestDTO redemptionRequestDTO, ComponentPreferenceDTO componentPreferenceDTO)

Table 2–63 (Cont.) paymentTransfer OBP Service Classes

OBP Application Service Name	TermDepositRedemptionApplicationService
Method name	inquireAndDoRedemption(SessionContext sessionContext ,RedemptionRequestDTO redemptionRequestDTO, ComponentPreferenceDTO componentPreferenceDTO)

2.22 save

Description:

This method is used to perform save operation.

Syntax:

```
public String save(List acctId, String partyId, String applicationId, String initiatedBy,
String reliefEffectiveDt,String reliefExpiryDt, String applicationDt, String startDt,
String endDt, String expiryDt, String approvedDate, String approvedBy)
```

Input:

Table 2–64 save Input Parameter

Parameter	Type	Description	Mandatory
acctId	String	Indicates the host account number.	Y
partyId	String	Indicates the host customer number.	Y
applicationId	String	Indicates the application ID.	Y
initiatedBy	String	Indicates the initiator name.	Y
reliefEffectiveDt	String	Indicates the effective date.	Y
reliefExpiryDt	String	Indicates the relief expiry date.	Y
applicationDt	String	Indicates the application date.	Y
startDt	String	Indicates the start date.	Y
endDt	String	Indicates the end date.	Y
expiryDt	String	Indicates the expiry date.	Y
approvedDate	String	Indicates the approved date.	Y
approvedBy	String	Indicates the approver name.	Y

Output:

Table 2–65 save Output Type

Type	Description
String	Returns null for success else returns error code.

OBP Service Name:

Table 2–66 save OBP Service Classes

NGP Adapter Class Name	HardshipReliefAssistanceApplyAdapter
Method Name	save(hardshipReliefRequestDTO)
OBP Application Service Name	HardshipReliefAssistanceApplicationService
Method name	save(fetchSessionContext(), hardshipReliefRequestDTO)

2.23 fetchDataFromHostSystem

Description:

This method is used to fetch data from the host system.

Syntax:

```
public HostEntityDataDTO fetchDataFromHostSystem(String hostCustomerNbr,
String sourceHostId, String inapplicableAccountStatusesForHardship, String
accrualStatusFlg, String assetClassCd, String usrDefnAcctStatusFlg, Boolean
fromPullNonDelinquent, String hostAcctNbr)
```

Input:

Table 2–67 *fetchDataFromHostSystem Input Parameter*

Parameter	Type	Description	Mandatory
hostCustomerNbr	String	Indicates the host customer number.	Y
sourceHostId	String	Indicates the source host ID.	Y
inapplicableAccountStatusesForHardship	String	Indicates the applicable account status.	Y
accrualStatusFlg	String	Indicates the accrual flag status.	Y
assetClassCd	String	Indicates the asset's class Cd.	Y
usrDefnAcctStatusFlg	String	Indicates the user definition account status flag.	
fromPullNonDelinquent	String	This switch indicates whether to be pulled from non delinquent or not.	
hostAcctNbr	String	Indicates the host account number.	

Output:

Table 2–68 *fetchDataFromHostSystem Output Type*

Type	Description
HostEntityDataDTO	Returns HostEntityDataDTO object.

OBP Service Name:

Table 2–69 *fetchDataFromHostSystem OBP Service Classes*

NGP Adapter Class Name	CollectionNGPAdapter.java
Method Name	fetchDataFromHostSystem
OBP Application Service Name	NA
Method name	NA

2.24 initiateRosoProcessForApprovalProcess

Description:

This method is used to create approval request for ROSO process.

Syntax:

public boolean initiateRosoProcessForApprovalProcess(RosoApprovalRequestDTO rosoApprovalRequestDTO) throws Exception

Input:

Table 2–70 *initiateProcessForApproval Input Parameter*

Parameter	Type	Description	Mandatory
rosoApprovalRequestDTO	com.splwg.ccb.domain.extension.adapter.dto.RosoApprovalRequestDTO	Indicates the DTO.	Y

Output:

Table 2–71 *initiateProcessForApproval Output Type*

Type	Description
boolean	Returns true for success else, return false.

OBP Service Name:

Table 2–72 *initiateProcessForApproval OBP Service Classes*

NGP Adapter Class Name	ROSOProcessForApprovalCallingAdapter
Method Name	callApproval(approvalRequestDTO)
OBP Application Service Name	ROSOProcessForApprovalApplicationService
Method name	callApproval(fetchSessionContext(), rosoProcessForApprovalRequestDTO)

2.25 getFactDetailsForRule

Description:

This method is used to retrieve all facts for a rule. The response object contains details, such as name and data type of each fact. RuleSet is not supported.

Syntax:

```
public CollRuleFactDetailsInquiryResponse getFactDetailsForRule(String ruleId,
String effectiveDate)
```

Input:

Table 2–73 *getFactDetailsForRule Input Parameter*

Parameter	Type	Description	Mandatory
ruleId	String	Rule ID for which fact information is to be retrieved.	Y
effectiveDate	String	Indicates the effective rule date.	N

Output:

Table 2–74 *getFactDetailsForRule Output Type*

Type	Description
CollRuleFactDetailsInquiryResponse	Response contains details of all facts for a Rule. CollRuleFactDetailsInquiryResponse object wraps list of CollFactDetailsDTO, which holds fact information. CollFactDetailsDTO contains name and dataType of fact.

OBP Service Name:

Table 2–75 *getFactDetailsForRule OBP Service Classes*

NGP Adapter Class Name	CollectionRulesAdapter
Method Name	getFactDetailsForRule(String ruleId,String effectiveDate)
OBP Application Service Name	RuleMaintenanceApplicationService
Method name	getFactDetailsForRule(SessionContext sessionContext, String ruleId, Date effectiveDate)

2.26 executeRuleWithPrePopulatedFactDetails

Description:

This method is used to execute rule or ruleset. It requires all required fact details for execution as input.

Syntax:

```
public Map<String, RuleFactInfoDto>
executeRuleWithPrePopulatedFactDetails(String ruleId, String ruleEffectiveDate,
List<RuleFactInfoDto> ruleFactInfoList, boolean isRuleSet)
```

Input:

Table 2–76 *executeRuleWithPrePopulatedFactDetails Input Parameter*

Parameter	Type	Description	Mandatory
ruleId	String	Indicates the rule or ruleset ID to be executed.	Y
ruleEffectiveDate	String	Indicates the rule or ruleset effective date.	N
ruleFactInfoList	List<RuleFactInfoDto>	Indicates the list of Rule/Ruleset facts required for execution.	Y
isRuleSet	Boolean	Boolean flag indicating if given id is of Rule/ RuleSet. Send true to execute rule set.	Y

Output:

Table 2–77 *executeRuleWithPrePopulatedFactDetails Output Type*

Type	Description
Map<String, RuleFactInfoDto>	Rule/RuleSet execution response. It contains response facts details. Key = fact name and Value = RuleFactInfoDto object holding facts details.

OBP Service Name:**Table 2–78 executeRuleWithPrePopulatedFactDetails OBP Service Classes**

NGP Adapter Class Name	CollectionRulesAdapter
Method Name 1	executeRule(String ruleId, String ruleEffectiveDate, FactInformationDTO[] factInformationDTOs)
OBP Application Service Name	RuleExecutionApplicationService
Method name	executeLatestRuleForGivenDate(SessionContext sessionContext, RuleContext ruleContext, Date executeForDate)
Method Name 2	executeRuleSet(String ruleSetId, String ruleEffectiveDate, FactInformationDTO[] factInformationDTOs)
OBP Application Service Name	RuleExecutionApplicationService
Method name	executeRuleSet(SessionContext sessionContext, RuleContext ruleContext)

2.27 manualAccountWriteOffProcess

Description:

This method is used to perform manual account write-off.

Syntax:

```
public TransactionStatus manualAccountWriteOffProcess(String accountNumber)
```

Input:**Table 2–79 manualAccountWriteOffProcess Input Parameter**

Parameter	Type	Description	Mandatory
accountNumber	String	Indicates the account number.	Y

Output:**Table 2–80 manualAccountWriteOffProcess Output Type**

Type	Description
TransactionStatus	Returns TransactionStatus object

OBP Service Name:**Table 2–81 manualAccountWriteOffProcess OBP Service Classes**

NGP Adapter Class Name	AccountWriteOffProcessCallingAdapter
Method Name	callAccountWriteOffProcess(accountNumber)
OBP Application Service Name	AccountWriteOffProcessApplicationService
Method name	callAccountWriteOffProcess(fetchSessionContext(), accountNumber)

2.28 updateCollateralRealizationStatus

Description:

This method is used to update collateral realization status.

Syntax:

```
public void updateCollateralRealizationStatus(CollateralStatusReaiizationUpdate_
DTO collateralStatusReaiizationUpdate_DTO) throws Exception
```

Input:

Table 2–82 updateCollateralRealizationStatus Input Parameter

Parameter	Type	Description	Mandatory
collateralStatusReaiizationUpdate_DTO	CollateralStatusReaiizationUpdate_DTO	Indicates the DTO.	Y

Output:

Table 2–83 updateCollateralRealizationStatus Output Type

Type	Description
void	No output

OBP Service Name:

Table 2–84 updateCollateralRealizationStatus OBP Service Classes

NGP Adapter Class Name	CollateralUpdateRealizationAdapter
Method Name	updateCollateralRealization(collateralrealizationdto)
OBP Application Service Name	CollateralRealizationApplicationService
Method name	updateCollateralRealization(fetchSessionContext(), collateralRealizationDTO)

2.29 abortCompositeInstancesByTitle

Description:

This method is used to abort composite instance by title, that is for cancelling the approval workflow.

Syntax:

```
public boolean abortCompositeInstancesByTitle(WorkflowManager_
DTO workflowManager_DTO)
```

Input:

Table 2–85 abortCompositeInstancesByTitle Input Parameter

Parameter	Type	Description	Mandatory
workflowManager_DTO	WorkflowManager_DTO	Indicates the DTO.	Y

Output:

Table 2–86 abortCompositeInstancesByTitle Output Type

Type	Description
boolean	Returns true for success else return false.

OBP Service Name:

Table 2–87 abortCompositeInstancesByTitle OBP Service Classes

NGP Adapter Class Name	WorkflowManagerCallingAdapter
Method Name	abortCompositeInstancesByTitle(workflowManagerDto)
OBP Application Service Name	WorkflowManagerApplicationService
Method name	abortCompositeInstancesByTitle(fetchSessionContext(), workflowManagerWrapperDTO)

2.30 sendMail

Description:

This method is used to send e-mails.

Syntax:

public TransactionStatus sendMail(Map mailParams) throws Exception

Input:

Table 2–88 sendMail Input Parameter

Parameter	Type	Description	Mandatory
mailParams	Java.util.Map	CollOugtBoundDocReqDto	Y

Output:

Table 2–89 sendMail Output Type

Type	Description
TransactionStatus	It returns TransactionStatus object.

OBP Service Name:

Table 2–90 sendMail OBP Service Classes

NGP Adapter Class Name	inCollectionAdapter
Method Name	sendEmail(adhocActionDTO)
OBP Application Service Name	ActionLogApplicationService
Method name	processAdhocAction(sessionContext, adhocData)

2.31 getFinancialDates

Description:

This method is used to fetch the financial dates.

Syntax:

```
public FinancialDatesDTO getFinancialDates(String bankCode, String branchCode)
throws Exception
```

Input:**Table 2–91** *getFinancialDates Input Parameter*

Parameter	Type	Description	Mandatory
bankCode	String	Bank code	Y
branchCode	String	Branch code	Y

Output:**Table 2–92** *getFinancialDates Output Type*

Type	Description
FinancialDatesDTO	Returns FinancialDatesDTO object, which contains the financial dates.

OBP Service Name:**Table 2–93** *getFinancialDates OBP Service Classes*

NGP Adapter Class Name	collectionDateManagerAdapter
Method Name	getFinancialDates(bankCode, branchCode)
OBP Application Service Name	CoreApplicationService
Method name	fetchBranchDates(sessionContext, bankCode, branchCode)

2.32 fetchTermDepositDetails

Description:

This method is used to fetch all associated deposits of a Term Deposit account.

Syntax:

```
public List<HostDepositDetailsDTO> fetchTermDepositDetails(String
accountNumber)
```

Input:**Table 2–94** *fetchTermDepositDetails Input Parameter*

Parameter	Type	Description	Mandatory
accountNumber	String	Term Deposit account number, whose associated deposit details need to be fetched	Y

Output:**Table 2–95** *fetchTermDepositDetails Output Type*

Type	Description
List<HostDepositDetailsDTO>	List of all associated Deposits of a Term Deposit account.

OBP Service Name:

Table 2–96 fetchTermDepositDetails OBP Service Classes

NGP Adapter Class Name	LinkedAccountBalanceFromTDAadapter
Method Name	fetchAllDepositDetails(String accountId, DepositStatus depositStatus)
OBP Application Service Name	DepositApplicationService
Method name	fetchAllDeposits(SessionContextsessionContext,String accountId,DepositStatus depositStatus)

2.33 fetchTDAccount

Description:

This method is used to fetch term deposit account details along with all its associated deposits.

Syntax:

```
public NgpAccountInformationBean fetchTDAccount(String accountNumber)
```

Input:

Table 2–97 fetchTDAccount Input Parameter

Parameter	Type	Description	Mandatory
accountNumber	String	Term Deposit account number, whose details need to fetched.	Y

Output:

Table 2–98 fetchTDAccount Output Type

Type	Description
NgpAccountInformationBean	Response containing Term Deposit account and its associated deposit details.

OBP Service Name:

Table 2–99 fetchTDAccount OBP Service Classes

NGP Adapter Class Name	LinkedAccountInfoFromTDAadapter
Method Name	fetchAccount(String accountNumber)
OBP Application Service Name	TermDepositAccountApplicationService
Method name	fetchAccount(SessionContext sessionContext,TermDepositAccountDTO termDepositAccountDTO)

2.34 initiateHardshipForApprovalProcess

Description:

This method is used to create approval request for hardship.

Syntax:

Public boolean initiateHardshipForApprovalProcess(HardshipApprovalRequestDTO hardshipApprovalRequestDTO)

Input:

Table 2–100 *initiateHardshipForApprovalProcess Input Parameter*

Parameter	Type	Description	Mandatory
hardshipApprovalRequestDTO	HardshipApprovalRequestDTO	HardshipApprovalRequest DTO	Y

Output:

Table 2–101 *initiateHardshipForApprovalProcess Output Type*

Type	Description
boolean	Returns true if success.

OBP Service Name:

Table 2–102 *initiateHardshipForApprovalProcess OBP Service Classes*

NGP Adapter Class Name	HardshipProcessForApprovalCallingAdapter
Method Name	callApproval(HardshipProcessForApprovalRequestDTO approvalRequestDTO)
OBP Application Service Name	HardshipProcessForApprovalApplicationService
Method name	callApproval(SessionContext sessionContext, HardshipApprovalRequestDTO hardshipProcessForApprovalRequestDTO)

2.35 updateIndividualDemographics

Description:

This method is used to modify individual party demographics details.

Syntax:

```
public boolean
updateIndividualDemographics(CollectionIndividualDemographicsDTO
collectionIndividualDemographicsDTO)
```

Input:

Table 2–103 *updateIndividualDemographics Input Parameter*

Parameter	Type	Description	Mandatory
collectionIndividualDemographicsDTO	CollectionIndividualDemographicsDTO	CollectionIndividualDemographicsDTO	Y

Output:

Table 2–104 *updateIndividualDemographics Output Type*

Type	Description
boolean	Returns true if success.

OBP Service Name:

Table 2–105 *updateIndividualDemographics OBP Service Classes*

NGP Adapter Class Name	CollectionPartyDemographicsAdapter
Method Name	updateIndividualDemographics (IndividualDemographicsDTO individualDemographicsDTO)
OBP Application Service Name	PartyDemographicsApplicationService
Method name	updateIndividualDemographics(SessionContext sessionContext, IndividualDemographicsDTO partyDemographicsDTO)

2.36 isWorkingDay

Description:

This method is used to validate the given date. It returns true if the given date is valid business date and returns false if the given date is not a business date or holiday.

Syntax:

```
public boolean isWorkingDay(Date date)
```

Input:

Table 2–106 *isWorkingDay Input Parameter*

Parameter	Type	Description	Mandatory
date	Date	Input date	Y

Output:

Table 2–107 *isWorkingDay Output Type*

Type	Description
boolean	Returns true if given date is valid business date and returns false if given date is not a business date or holiday

OBP Service Name:

Table 2–108 *isWorkingDay OBP Service Classes*

NGP Adapter Class Name	CollectionDateManagerDateAdapter
Method Name	isWorkingDay(Date date)
OBP Adapter name	ICalendarServiceAdapter.
Method name	isValidPostingDate(p_ SessionContext.getTransactionBranch(),inputDate))

2.37 deleteWorklistItemForApplicationId

Description:

This method is used to delete the work list item for a give application and party.

Syntax:

```
public void deleteWorklistItemForApplicationId(String applicationId, String partyId)
```

Input:

Table 2–109 deleteWorklistItemForApplicationId Input Parameter

Parameter	Type	Description	Mandatory
applicationId	String	Application id	Y
partyId	String	Party Id	Y

Output:

Table 2–110 deleteWorklistItemForApplicationId Output Type

Type	Description
Void	

OBP Service Name:

Table 2–111 deleteWorklistItemForApplicationId OBP Service Classes

NGP Adapter Class Name	HardshipReliefRequestAdapter
Method Name	deleteWorklistItemForTransactionReference(String applicationId, String partyId)
OBP Application Service Name	WorkListApplicationService
Method name	deleteWorkItemForTransactionReferenceNumber(varWorkItemDTO.getId())

