**REFUND BILL PAYMENTS**

Submitted to



**On**

**27-Feb-2024**

**By**



**Best of Breed Software Solutions (India) Pvt Limited**

**BBSS House**

**1st Main, 1st Block, BTM IV stage,**

**Bangalore 560 076**

|  |  |  |  |
| --- | --- | --- | --- |
| Document No. | Finacle EBanking Customization | Version No. | 1.00 |
| Prepared By | Jagannath Swain | Signature/  Date | 27-Feb-24 |
| Verified By | Arun Vasan R |  |  |

**Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Version History** | **Reviewed by** |
| 1.00 | 27-Feb-24 | Original version | Arun Vasan R |

The information and data contained in this document is confidential to Best of Breed Software Solutions India Private Limited and Nainital Bank. This shall not be disclosed to a third party without prior written consent of Best of Breed Software Solutions (India) Private Limited.

**Table of Contents**

**1.** Refund Success **………………………………………………..……………….4**

**2.** Partial Refund (Success)**……………………………………………………...….5**

**3.** Refund Exceeds Transaction Amount **……………………………….………….....7**

**4.** Duplicate Refund (Fail)**………………………………………………………….8**

# 5. Transaction not found……………………………………………………………9

# 6. Technical Issue (Pending Status)………………………………………………………………………………10

# 7. Retry (Technical Issue refunds)…………………………………………..…………………………………11

**8. Sources……………………………………………………………………...12**

[8.1 Type System and Value Objects](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945456)**………………………………………………....12**

[8.2 Formgroup5](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945457)**………………………………………………………………….12**

[8.3 HIF](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945458)**….………………………………………………………………..….....12**

[8.4 Data Files](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945459)**…………………………………………………………………...12**

[8.5 Service Files](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945461)**…………………………………………………………………13**

[8.6 Java Files](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945462)**…………………………….……………………………...............13**

8.7 CLASS FILES**…………………………….……………………………...........14**

[8.8 JSP/Styles/JS Files](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945463)**……………………………………………………………16**

[8.9 Database Configuration](file:///F:\NTB_ReleaseNotes_EBK_233.docx#_Toc11945465) **………………….……………………………...........16**

1. **Refund Success**

Refund API is an API used to initiate a real-time refund/ cancellation, full/ partial against a transaction which is initiated in the past.

It can be used multiple times against a particular transaction

During the Refund API invoke time bank will send encrypted request, for which first bank will decrypt the

Encrypt request using this REFUND\_AES\_SECRET\_KEY (ABCDEF0123456789ABCDEF0123456789) received at bank end and after that based on decrypted request.

**Note:**

**Before sending the request Bank have to make sure Merchant client IP and Payee ID should be present on this custom table CUSTOM\_REFUNDS\_IP\_CONFIG\_TABLE (CRIP) then only bank request should be process.**

**Sample Request for Encrypt data:**

**{**

**"encData": "39EC33E6EF014CE89BBBB020DD199826987F82DA7C12CD4751A9DCB8E3A80DC167E9622291D10FF61DC7918B67137FD4B15563B8E198DA7D56D35566BAE50F1FF1B65BAC96CEDE623AED205D36EE2CE2876656CC2DDBF30C90DEBD18D309549E",**

**"channelID": "BILLPAYMENT"**

**}**

Sample Response for Encrypt data:

**{"encData":"0B195E871B91644E202448F06B78770DCB6EDCB8FEC0D60ABB0CDB932199E342","responseCode":"00"}**

**Sample Request:**

**{"MerchantRefNo": "100036360", "MerTxnRefNo": "208500","RefundAmount": "50.00" }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036360 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 50.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036360","MerTxnRefNo" : "208500","RefundAmount" : "50.00","BankRefNo" : "100526","BankTxnRefNo" : "856595","ErrorCode" : "00","ErrorDescription" : "Refund successfully processed" }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036360 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 50.00 |
| BankRefNo | Bank Refund Reference Number | 100526 |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856595 |
| ErrorCode | Error\_Code | 00 |
| ErrorDescription | Error\_Description | Refund successfully processed |

Refund API should validate whether Total refunds initiated (including the current one) is always less than or equal to transaction amount of underlying transaction o Ex: Bill Payment has completed a transaction with transaction amount as Rs.100 and Bill Payment transaction reference number as ‘208500’.

# 2. Partial Refund (Success)

Bill Payment fires a refund with Bill Payment refund reference number as ‘100036360’, Bill Payment transaction refund reference number as ‘208500’ and refund amount as Rs.50 – Refund API should allow this request, process it successfully and respond back to Bill Payment real-time

**Sample Request for Encrypt data:**

**{**

**"encData": "39EC33E6EF014CE89BBBB020DD199826987F82DA7C12CD4751A9DCB8E3A80DC167E9622291D10FF61DC7918B67137FD4B15563B8E198DA7D56D35566BAE50F1FF1B65BAC96CEDE623AED205D36EE2CE2876656CC2DDBF30C90DEBD18D309549E",**

**"channelID": "BILLPAYMENT"**

**}**

**Sample Response for Encrypt data:**

**{"encData":"** **2347BFC929D230AFE512E4C01EECAC68D7B82FEC535341832A35E925551633BED9A14A349E08AF77084E2741AC94217F4E0D031D5019D436C359AACAA6AE492D**

**","responseCode":"00"}**

**Sample Request:**

**{"MerchantRefNo": "100036360", "MerTxnRefNo" : "208500","RefundAmount" : "50.00" }**

|  |  |  |
| --- | --- | --- |
| Field Name for checksum generation | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036360 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 50.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036360","MerTxnRefNo" : "208500","RefundAmount" : "50.00","BankRefNo" : "100527","BankTxnRefNo" : "856597","ErrorCode" : "00","ErrorDescription" : "** **Duplicate request – Refund successfully processed previously" }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036360 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 50.00 |
| BankRefNo | Bank Refund Reference Number | 100527 |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856597 |
| ErrorCode | Error\_Code | 00 |
| ErrorDescription | Error\_Description | Duplicate request – Refund successfully processed previously |

# 3. Refund Exceeds Transaction Amount

Bill Payment fires another refund with Bill Payment refund reference number as ‘100036361’, Bill Payment transaction refund reference number as ‘208500’ and refund amount as Rs.51. Total refund amount including this refund against this transaction will be Rs.101 (Rs. 50 + Rs. 51) and hence this refund should not be allowed by the API and fail with appropriate reason

**Sample Request:**

**{"MerchantRefNo": "100036361", "MerTxnRefNo": "208500","RefundAmount": "51.00"}**

|  |  |  |
| --- | --- | --- |
| Field Name for checksum generation | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036361 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 51.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036361","MerTxnRefNo" : "208500","RefundAmount" : "50.00","BankRefNo" : "NA", "BankTxnRefNo" : "856595","ErrorCode" : "01","ErrorDescription" : "Refund failed as total refunds are exceeding transaction amount" }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036361 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 51.00 |
| BankRefNo | Bank Refund Reference Number | NA |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856595 |
| ErrorCode | Error\_Code | 01 |
| ErrorDescription | Error\_Description | Refund failed as total refunds are exceeding transaction amount |

# 4. Duplicate Refund (Fail)

Refund API should not allow duplicate refunds (duplicity should be checked basis merchant refund reference number) o In the above example, if Bill Payment fires a refund again with Bill Payment refund reference number as ‘100036361’ then system should not process the refund and respond back to Bill Payment as ‘Refund already processed’

**Sample Request:**

**{"MerchantRefNo": "100036361", "MerTxnRefNo" : "208500","RefundAmount" : "51.00"}**

|  |  |  |
| --- | --- | --- |
| Field Name for checksum generation | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036361 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 51.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036361","MerTxnRefNo" : "208500","RefundAmount" : "50.00","BankRefNo" : "NA", "BankTxnRefNo" : "856595","ErrorCode" : "01","ErrorDescription" : "Refund failed as total refunds are exceeding transaction amount " }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036361 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208500 |
| RefundAmount | Refund Amount | 51.00 |
| BankRefNo | Bank Refund Reference Number | NA |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856595 |
| ErrorCode | Error\_Code | 01 |
| ErrorDescription | Error\_Description | Duplicate request – Refund failed as total refunds are exceeding transaction amount |

# 5. Transaction not found

**Sample Request:**

**{"MerchantRefNo": "100036362", "MerTxnRefNo": "208501","RefundAmount": "50.00" }**

|  |  |  |
| --- | --- | --- |
| Field Name for checksum generation | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036362 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208501 |
| RefundAmount | Refund Amount | 50.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036362","MerTxnRefNo" : "208501","RefundAmount" : "50.00","BankRefNo" : "NA", "BankTxnRefNo" : "NA", "ErrorCode" : "01","ErrorDescription" : " Transaction not found "}**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036362 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208501 |
| RefundAmount | Refund Amount | 50.00 |
| BankRefNo | Bank Refund Reference Number | NA |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | NA |
| ErrorCode | Error\_Code | 01 |
| ErrorDescription | Error\_Description | Transaction not found |

# 6. Technical Issue (Pending Status)

**Sample Request:**

**{"MerchantRefNo": "100036363", "MerTxnRefNo": "208503","RefundAmount": "50.00" }**

|  |  |  |
| --- | --- | --- |
| Field Name for checksum generation | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036363 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208503 |
| RefundAmount | Refund Amount | 50.00 |

**Sample Response:**

**{ "MerchantRefNo": "100036363","MerTxnRefNo": "208503","RefundAmount" : "50.00","BankRefNo" : "100527","BankTxnRefNo" : "856597","ErrorCode" : "02","ErrorDescription" : " Technical issue. Please try again. "}**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036363 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208503 |
| RefundAmount | Refund Amount | 50.00 |
| BankRefNo | Bank Refund Reference Number | 100527 |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856597 |
| ErrorCode | Error\_Code | 02 |
| ErrorDescription | Error\_Description | Technical issue. Please try again. |

# 7. Retry (Technical Issue refunds)

**Sample Request:**

**{"MerchantRefNo": "100036363", "MerTxnRefNo" : "208503","RefundAmount" : "50.00" }**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036363 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208503 |
| RefundAmount | Refund Amount | 50.00 |

**Sample Response:**

**{ "MerchantRefNo" : "100036363","MerTxnRefNo" : "208503","RefundAmount" : "50.00","BankRefNo" : "100527","BankTxnRefNo" : "856597","ErrorCode" : "00","ErrorDescription" : " Refund processed successfully "}**

|  |  |  |
| --- | --- | --- |
| Field Name | Description | Sample |
| MerchantRefNo | Merchant Refund Reference Number | 100036363 |
| MerTxnRefNo | Merchant Transaction Reference Number | 208503 |
| RefundAmount | Refund Amount | 50.00 |
| BankRefNo | Bank Refund Reference Number | 100527 |
|  |  |  |
| BankTxnRefNo | Bank Transaction Reference Number | 856597 |
| ErrorCode | Error\_Code | 00 |
| ErrorDescription | Error\_Description | Refund processed successfully |

# Sources

This section has list of source files added or customized for this module. The source files are provided as part of the release.

## 8.1 Type System and Value Objects

|  |  |  |
| --- | --- | --- |
| Sr. No. | File Name | Path |
| 1. | CustomRefundsDetailsVO.xml | \spec\vo |
| 2. | CustomRefundsDetailsVO.java | \src\com\ntb\ebanking\types\valueobjects |
| 3. | ICustomRefundsDetailsVO.java | \src\com\ntb\ebanking\types\valueobjects |
| 4. | VOFactory.java | \src\com\ntb\ebanking\types\valueobjects |

## Formgroup

Unavailability of files:

## HIF

Unavailability of files:

## Data Files

### 

| Sr. No. | File Name | Path |
| --- | --- | --- |
| 1. | DAL.properties | \WorkingDir\data |
| 2. | CustomTypesCatalogue.xml | \WorkingDir\data |

## Service Files

| Sr. No. | File Name | Path |
| --- | --- | --- |
| 1. | CustomRefundAccBalService.xml | \src |
| 2. | web.xml | src\WebContent\WEB-INF |

## 8.6 Java Files

| Sr. No. | File Name | Path |
| --- | --- | --- |
| 1. | CustomBillPaymentEncryptionUtil.java | \src\com\ntb\ebanking\ billPayments \ util |
| 2. | CustomBillPaymentUtility.java | \src\com\ntb\ebanking\ billPayments \ util |
| 3. | CustomRecordFromCpayDataBuilder.java | \src\com\ntb\ebanking\ dbaccess\dal\ databuilder |
| 4. | CustomRecordFromTXNDDataBuilder.java | \src\com\ntb\ebanking\ dbaccess\dal\ databuilder |
| 5. | CustomRefundAmountDataBuilder.java | \src\com\ntb\ebanking\dbaccess\dal\ databuilder |
| 6. | CustomRecordFromCpayQueryBuilder.java | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 7. | CustomRecordFromTXNDQueryBuilder.java | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 8. | CustomRefundAmountQueryBuilder.java | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 9 | CustomBBPSRefundsAmountInputVO.java | \src\com\ntb\ebanking\refunds\controller |
| 10. | CustomBillPaymentPaymentsRefundsController.java | \src\com\ntb\ebanking\refunds\controller |
| 11 | CustomRefundAccBalServiceRefundAmountFromTxndImpl.java | \src\com\ntb\ebanking\refunds\service |
| 12. | CRAPDAO.java | \ src\com\ntb\ebanking\ tao \ dao |
| 13. | CRAPInfo.java | \src\com\ntb\ebanking\ tao \info |
| 14. | CRAPInfoKey.java | \src\com\ntb\ebanking\ tao\infokey |
| 15. | CRAPConstants.java | \src\com\ntb\ebanking\ tao |
| 16. | CRAPTAO.java | \src\com\ntb\ebanking\ tao |
| 17. | CookieConstants.java | \src\com\ntb\ebanking\ tao |
| 18. | CustomRefundsDetailsVO.java | \src\com\ntb\ebanking\ types\valueobjects |
| 19. | ICustomRefundsDetailsVO.java | \src\com\ntb\ebanking\ types\valueobjects |
| 20. | CustomTypesCatalogueConstants.java | \src\com\ntb\ebanking\ types\valueobjects |
| 21. | VOFactory.java | \src\com\ntb\ebanking\ types\valueobjects |
| 22. | CustomRefundsIPConfig.java | \src\com\ntb\ebanking\ types\valueobjects |
| 23. | ICustomRefundsIPConfig.java | \src\com\ntb\ebanking\ types\valueobjects |
| 24. | CustomRecordFromCripDataBuilder.java | \src\com\ntb\ebanking\dbaccess\dal\ databuilder |
| 25. | CustomRecordFromCripQueryBuilder.java | \src\com\ntb\ebanking\dbaccess\dal\Querybuilder |

## 8.7 Class Files

|  |  |  |
| --- | --- | --- |
| Sr. No | File Name | Path |
| 1. | CustomBillPaymentEncryptionUtil.class | \src\com\ntb\ebanking\ billPayments \ util |
| 2. | CustomBillPaymentUtility.class | \src\com\ntb\ebanking\ billPayments \ util |
| 3. | CustomRecordFromCpayDataBuilder.class | \src\com\ntb\ebanking\ dbaccess\dal\ databuilder |
| 4. | CustomRecordFromTXNDDataBuilder.class | \src\com\ntb\ebanking\ dbaccess\dal\ databuilder |
| 5. | CustomRefundAmountDataBuilder.class | \src\com\ntb\ebanking\dbaccess\dal\ databuilder |
| 6. | CustomRecordFromCpayQueryBuilder.class | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 7. | CustomRecordFromTXNDQueryBuilder.class | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 8. | CustomRefundAmountQueryBuilder.class | \src\com\ntb\ebanking\dbaccess\dal \ querybuilder |
| 9 | CustomBBPSRefundsAmountInputVO.class | \src\com\ntb\ebanking\refunds\controller |
| 10. | CustomBillPaymentPaymentsRefundsController.class | \src\com\ntb\ebanking\refunds\controller |
| 11 | CustomRefundAccBalServiceRefundAmountFromTxndImpl.class | \src\com\ntb\ebanking\refunds\service |
| 12. | CRAPDAO.class | \ src\com\ntb\ebanking\ tao \ dao |
| 13. | CRAPInfo.class | \src\com\ntb\ebanking\ tao \info |
| 14. | CRAPInfoKey.class | \src\com\ntb\ebanking\ tao\infokey |
| 15. | CRAPConstants.class | \src\com\ntb\ebanking\ tao |
| 16. | CRAPTAO.class | \src\com\ntb\ebanking\ tao |
| 17. | CookieConstants.class | \src\com\ntb\ebanking\ tao |
| 18. | CustomRefundsDetailsVO.class | \src\com\ntb\ebanking\ types\valueobjects |
| 19. | ICustomRefundsDetailsVO.class | \src\com\ntb\ebanking\ types\valueobjects |
| 20. | CustomTypesCatalogueConstants.class | \src\com\ntb\ebanking\ types\valueobjects |
| 21. | VOFactory.class | \src\com\ntb\ebanking\ types\valueobjects |
| 22. | CustomRefundsIPConfig.class | \src\com\ntb\ebanking\ types\valueobjects |
| 23. | ICustomRefundsIPConfig.class | \src\com\ntb\ebanking\ types\valueobjects |
| 24. | CustomRecordFromCripDataBuilder.class | \src\com\ntb\ebanking\dbaccess\dal\ databuilder |
| 25. | CustomRecordFromCripQueryBuilder.class | \src\com\ntb\ebanking\dbaccess\dal\Querybuilder |

## 8.8 JSP/Styles/JS Files

Unavailability of files:

## 8.9 Database Configuration

| Sr. No. | Table Name | Queries |
| --- | --- | --- |
| 1. | CRAP | CREATE TABLE CUSTOM\_REFUNDS\_PAYMENT\_TABLE  (  DB\_TS NUMBER(5),  BANK\_ID NVARCHAR2(11),  USER\_ID NVARCHAR2(32),  CUST\_ID NVARCHAR2(200),  MER\_REF\_NO NVARCHAR2(65),  MER\_TXN\_ID NVARCHAR2(65),  REFUND\_AMOUNT NUMBER(18,3),  TOTAL\_TXN\_AMOUNT NUMBER(18,3),  REST\_AMOUNT NUMBER(18,3),  BANK\_REF\_NO NVARCHAR2(65),  BANK\_TXN\_REF\_NO NVARCHAR2(65),  TXN\_CURRENCY NVARCHAR2(3),  ACCOUNT\_NUMBER NVARCHAR2(35),  TXN\_DATE DATE,  REFUND\_DATE DATE,  BILLER\_NAME NVARCHAR2(140),  BILLER\_ID NVARCHAR2(32),  BILLPAYMENT\_TRAN\_NO NVARCHAR2(50),  PAYMENT\_TYPE NVARCHAR2(15),  PAYMENT\_MODE NVARCHAR2(20),  STATUS NVARCHAR2(32),  ERROR\_CODE NVARCHAR2(40),  ERROR\_DESC NVARCHAR2(256),  DEL\_FLG CHAR(1),  R\_MOD\_ID NVARCHAR2(65),  R\_MOD\_TIME DATE,  R\_CRE\_ID NVARCHAR2(65),  R\_CRE\_TIME DATE,  FREE\_FIELD\_1 NVARCHAR2(2000),  FREE\_FIELD\_2 NVARCHAR2(2000),  FREE\_FIELD\_3 DATE,  FREE\_FIELD\_4 DATE    );  CREATE SYNONYM CRAP FOR CUSTOM\_REFUNDS\_PAYMENT\_TABLE; |
| 2. | CLMT | Insert into CLMT (DB\_TS,BANK\_ID,TXN\_TYPE,SCHEME\_CODE,LIMIT\_TYPE,CHANNEL\_ID,MIN\_ENTRY\_LIMIT,MAX\_ENTRY\_LIMIT,CRNCY,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','BDK','DRL','E','I',1,999999999999,'INR','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into CLMT (DB\_TS,BANK\_ID,TXN\_TYPE,SCHEME\_CODE,LIMIT\_TYPE,CHANNEL\_ID,MIN\_ENTRY\_LIMIT,MAX\_ENTRY\_LIMIT,CRNCY,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','BDK','NCE','A','I',1,999999999999,'INR','N','01.SUMIT',sysdate,'01.SUMIT',sysdate);  Insert into CLMT (DB\_TS,BANK\_ID,TXN\_TYPE,SCHEME\_CODE,LIMIT\_TYPE,CHANNEL\_ID,MIN\_ENTRY\_LIMIT,MAX\_ENTRY\_LIMIT,CRNCY,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (2,'01','BDK','NCE','E','I',1,2000000,'INR','N','01.SUMIT',sysdate,'01.ADMIN2',sysdate);  Insert into CLMT (DB\_TS,BANK\_ID,TXN\_TYPE,SCHEME\_CODE,LIMIT\_TYPE,CHANNEL\_ID,MIN\_ENTRY\_LIMIT,MAX\_ENTRY\_LIMIT,CRNCY,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','BDK','NRE','E','I',1,999999999999,'INR','N','01.HEM',sysdate,'01.HEM',sysdate); |
| 3. | COCD | Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','AID','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','EFT','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','NFT','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','NTY','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','NWK','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','NWL','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','NWR','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','TPE','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','TTG','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','TTP','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate);  Insert into COCD (DB\_TS,BANK\_ID,CODE\_TYPE,CM\_CODE,LANG\_ID,CD\_DESC,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','TXN','BDK','001','Bill Desk','N','custom',sysdate,'custom',sysdate); |
| 4. | FCOT | Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','INITIATE','BDK','BDKBANKTYPE','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','INITIATE','BDK','BDKACCTYPE','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','INITIATE','BDK','BDKINCURRENCYFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','TRAN\_TYPE\_ATT\_NETWORK\_TYPES','BDK','BDKNETWORKFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','TRAN\_TYPE\_ATT\_PMT\_FREQ','BDK','BDKPMTFREQFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','TRAN\_TYPE\_ATT\_FREQ\_TYPE','BDK','BDKFREQTYPEFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','TRAN\_TYPE\_ATT\_ACCT\_IDENTFIER','BDK','BDKACCTIDENTIFIERFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','TRAN\_TYPE\_ATT\_CP\_TYPE','BDK','BDKCPTYPEFILTER','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','CPACCTYPE','BDK','BDKCPACCTYPE','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','CPBANKTYPE','BDK','BDKCPBANKTYPE','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','CPREGION','BDK','BDKCPREGION','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FCOT (DB\_TS,BANK\_ID,ENTITY,SCENARIO,FILTER\_NAME,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (2,'01','TRAN\_TYPE\_ATT\_CUR','BDK','BDKCURRENCYFILTER','N','01.HEM',sysdate,'01.HEM',sysdate); |
| 5. | FDET | Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKBANKTYPE','Bank Type Filter','BankTypeIndicator','IN','INT','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKACCTYPE','AccTypeIndicator','AccTypeIndicator','IN','SBA|CAA|ODA|CCA','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKINCURRENCYFILTER','INITIATOR CURRENCY FILTER','CM\_CODE','IN','INR','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKNETWORKFILTER','NETWORK FILTER NAME','CM\_CODE','IN','WIB','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','BDKPMTFREQFILTER','PAYMENT FREQUENCY FILTER NAME','IDCODE','IN',null,'N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKFREQTYPEFILTER','FREQUENCY TYPE FILTER NAME','IDCODE','IN','O','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKACCTIDENTIFIERFILTER','ACCOUNT IDENTIFIER FILTER NAME','IDCODE','IN','A','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKCPTYPEFILTER','BENEFICIARY FILTER NAME','IDCODE','IN','M','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKCPACCTYPE','COUNTER PARTY ACCOUNT TYPE FILTER','CPAccType','IN','CCD|TDA|MTF|LAA|SBA|CAA|ODA|CCA','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKCPBANKTYPE','BANK TYPE FILTER','CPBankType','IN','HBK|OBK|ABK','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (3,'01','BDKCPREGION','COUNTER PARTY REGION FILTER','CPRegion','IN','DOM|INT|SEP','N','01.HEM',sysdate,'01.HEM',sysdate);  Insert into FDET (DB\_TS,BANK\_ID,FILTER\_NAME,FILTER\_DESCRIPTION,ATTRIBUTE\_NAME,OPERATOR,FILTER\_VALUES,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (2,'01','BDKCURRENCYFILTER','CURRENCY FILTER NAME','CM\_CODE','IN','INR','N','01.HEM',sysdate,'01.HEM',sysdate); |
| 6. | PRPM | Insert into PRPM  (DB\_TS, BANK\_ID, APP\_ID, PROPERTY\_NAME, PROPERTY\_VAL, PROPERTYDESCRIPTION, DEL\_FLG, R\_MOD\_ID, R\_MOD\_TIME, R\_CRE\_ID, R\_CRE\_TIME)  Values  (1, '01', 'BWY', 'REFUND\_AES\_SECRET\_KEY', 'ABCDEF0123456789ABCDEF0123456789', 'Encryption Key', 'N', 'setup', SYSDATE, 'setup', SYSDATE);    Insert into PRPM  (DB\_TS, BANK\_ID, APP\_ID, PROPERTY\_NAME, PROPERTY\_VAL, PROPERTYDESCRIPTION, DEL\_FLG, R\_MOD\_ID, R\_MOD\_TIME, R\_CRE\_ID, R\_CRE\_TIME)  Values  (1, '01', 'BWY', 'BBPS\_TXN\_TYPE', 'BDK', 'BBPS Transaction type', 'N', 'setup', SYSDATE, 'setup', SYSDATE);        Insert into PRPM  (DB\_TS, BANK\_ID, APP\_ID, PROPERTY\_NAME, PROPERTY\_VAL, PROPERTYDESCRIPTION, DEL\_FLG, R\_MOD\_ID, R\_MOD\_TIME, R\_CRE\_ID, R\_CRE\_TIME)  Values  (1, '01', 'BWY', 'BBPS\_CHECKSUM\_SECRET\_KEY', 'TEST123', 'BBPS Checksum Secret Key type', 'N', 'setup', SYSDATE, 'setup', SYSDATE);        Insert into PRPM (DB\_TS,BANK\_ID,APP\_ID,PROPERTY\_NAME,PROPERTY\_VAL,PROPERTYDESCRIPTION,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME) values (1,'01','BWY','BDSK\_BBPS\_OTP\_MAX\_ATTEMPTS','3','OTP max attempts','N','setup',sysdate,'setup',sysdate); |
| 7. | TNTP | Insert into TNTP (BANK\_ID,TXN\_TYPE,DB\_TS,TXN\_CATEGORY,COUNTER\_PARTY\_ID,CONSUMPTION\_LIMIT\_APPLICABLE,MAX\_NO\_OF\_ENTRIES\_IN\_A\_TXN,HOT\_PAYMENT\_ALLOWED,SCHEDULING\_ALLOWED,RECURRING\_ALLOWED,STORED\_CENTRALLY,TERMS\_AND\_COND\_FLAG,EXTRA\_TRANSACTION\_INFO,MAX\_SCHEDULING\_PERIOD,TEMPLATE\_SUPPORTED,TRANSACTING\_ENTITY,NOTIFICATION\_REQUIRED,INIT\_ACCT\_SAME\_IN\_ALL\_ENTRIES,IS\_INPUT\_DATE\_EFFECTIVE\_DATE,TXN\_DEFERRED\_DAYS,ADV\_NO\_OF\_DAYS\_FOR\_PROCESSING,ORIGINATOR\_IDENTIFIER,IS\_ADDENDA\_REQD,DEL\_FLG,R\_CRE\_ID,R\_CRE\_TIME,R\_MOD\_ID,R\_MOD\_TIME,DOC\_UPLOAD\_FLAG,CONFIDENTIALITY\_SUPPORTED) values ('01','BDK',3,'P','41475','N',1,'Y','N','N','N','N', EMPTY\_CLOB(),10,'N','A','Y','N','N',0,0,null,'N','N','01.HEM',sysdate,'01.HEM',sysdate,'N','N'); |
| 8. | CRIP | CREATE TABLE CUSTOM\_REFUNDS\_IP\_CONFIG\_TABLE  (  DB\_TS NUMBER(5),  BANK\_ID NVARCHAR2(11),  CLINT\_IP\_CONFIG NVARCHAR2(65),  PAYEE\_ID NVARCHAR2(65),  DEL\_FLG CHAR(1),  R\_MOD\_ID NVARCHAR2(65),  R\_MOD\_TIME DATE,  R\_CRE\_ID NVARCHAR2(65),  R\_CRE\_TIME DATE,  FREE\_FIELD\_1 NVARCHAR2(2000),  FREE\_FIELD\_2 NVARCHAR2(2000),  FREE\_FIELD\_3 DATE,  FREE\_FIELD\_4 DATE  );  CREATE SYNONYM CRIP FOR CUSTOM\_REFUNDS\_IP\_CONFIG\_TABLE;  INSERT INTO CUSTOM\_REFUNDS\_IP\_CONFIG\_TABLE ( DB\_TS,BANK\_ID,CLINT\_IP\_CONFIG,PAYEE\_ID,DEL\_FLG,R\_MOD\_ID,R\_MOD\_TIME,R\_CRE\_ID,R\_CRE\_TIME,FREE\_FIELD\_1,FREE\_FIELD\_2,FREE\_FIELD\_3,FREE\_FIELD\_4 )  VALUES ( '0' , '01' , '10.212.128.2' , '41475' , 'N' , 'CUSTOM' , SYSDATE , 'CUSTOM' , SYSDATE , NULL , NULL , NULL , NULL ) |
|  |  |  |