

Oracle FLEXCUBE Direct Banking

System Handbook

Volume VI – Originations and Peer To Peer
Payments

Release 12.0.2.0.0

Part No. E50108-01

September 2013

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax: +91 22 6718 3001

www.oracle.com/financialservices/

Copyright © 2008, 2013, Oracle and/or its affiliates. All rights reserved.

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

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

Contents

1. Preface	1
1.1. Intended Audience.....	1
1.2. Documentation Accessibility.....	1
1.3. Access to OFSS Support	1
1.4. Structure	1
1.5. Related Information Sources	2
2. Abbreviations	3
3. Channel Enrollment	4
4. Channel On-boarding.....	7
5. Forgot Password	13
6. Security Questions	15
7. Origination Non BPEL and BPEL and Origination Leads.....	17
7.1. BPEL Configuration for STP	34
7.2. Origination leads.....	38
8. Peer to Peer Payments.....	45
9. Verification Framework	48

1. Preface

1.1. Intended Audience

This System Handbook VI – Originations and Peer to Peer Payments is intended for the following audience:

This document is primarily targeted at

- Application Architects
- End to End Designers
- Business Service Detailed Designers and Developers
- Implementation Partners

1.2. Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

1.3. Access to OFSS Support

<https://flexsupp.oracle.com/>

1.4. Structure

This document consists of the following chapter

Chapter 3, "Channel Enrollment"

This chapter discusses the Channel Enrollement System Parameters and its configuration.

Chapter 4, "Channel On boarding"

This chapter discusses the Channel On boarding System Parameters and its configuration.

Chapter 5, "Forgot Password"

This chapter discusses the Forgot Password System Parameters and its configuration.

Chapter 6, "Security Questions"

This chapter discusses the Security Questions System Parameters and its configuration.

Chapter 7, "Originations"

This chapter discusses the Origination System Parameters and its configuration.

Chapter 8, "Peer to Peer Payments"

This chapter discusses the Peer to Peer Payment System Parameters and its configuration.

Chapter 9, "Verification Framework"

This chapter discusses the Verification Framework System Parameters and its configuration.

1.5. Related Information Sources

For more information on Oracle FLEXCUBE Direct Banking Release 12.0.2.0.0., refer to the following documents:

- Oracle FLEXCUBE Direct Banking System Handbook – Volume I
- Oracle FLEXCUBE Direct Banking System Handbook – Volume II
- Oracle FLEXCUBE Direct Banking System Handbook – Volume III
- Oracle FLEXCUBE Direct Banking System Handbook – Volume IV
- Oracle FLEXCUBE Direct Banking System Handbook – Volume V

2. Abbreviations

FCDB	Oracle FLEXCUBE Direct Banking
HTTP	Hyper Text Transfer Protocol
J2EE	Java 2 Enterprise Edition
WL	Weblogic 11g
STP	Straight Through Process
BPEL	Business Process Execution Language

3. Channel Enrollment

This transaction allows a visiting user to enroll for default services of channels. The following configurations are required to enable creation of such a user.

A new user type, ENC (Enrolled User), has been introduced to cater to such users who do not have a banking relationship with the bank but would like to enroll with the bank for basic services. ENC user type is introduced to represent the prospect into FCDB application. This user type is configured so that any user created under this user type would not have a customer id assigned.

Channel Enrollment functionality has been given on login page with the hyper link.

No	Property Name	Property Value	Day-0/Day-1	Table/File	Description
1	<id_entity>.COE.SKIP_CAPCTHA_VALIDATION	FALSE/TRUE	Day 1	mstproperties	Based on property, while enrollment it will skip image verification. If flag set to false, one verification image will be displayed to customer. User has to manually enter text shown in the image within expiry time of that password
2	<id_entity>.ISSERVICE_REQ.<usertype>	FALSE/TRUE	Day 1	mstproperties	While enrollment user should be created through service request or not. If flag set to false then user is created without intervention of administrator. And if flag set to true then process is registered as service request and admin has to process the same using "Request processing"

					transaction. This will be used for both EN1 and ENC
3	<id_entity>.DEFAULT _ACTIVATE.<usertype >	Y/N	DAY 1	mstprop erties	While enrollment user should be created and default activated or not. If flag set to true then all users of those entity user types are created and automatically activated, if flag set to false, then user is created with deactivated state. Administrator has to activate user using "Activate User" transaction in user management. This is applicable for Retail User and Prospect user
4	<id_entity>.usertype >.DUMMYUSERID	Valid User ID of that user type	Day 1	mstprop erties	While enrollment if Service Request flag is set to true then service request will be created using this dummy user id. As it is non login transaction and user id is not created for customer at this point, we need dummy user id to create service request. This will be used for both EN1 and ENC
5	<id_entity>.<typecon tact>.COE.SKIP_CONT ACT_VERIFICATION	FALSE/TRUE	Day 1	mstprop erties	During enrollment verification of particular contact type of customer like email, mobile number can be skipped. If flag us set to true for one contact type, that contact will not be verified during the process. And if flag is set to false or property is not present then customer has

					to verify that contact type before proceeding or he will be granted limited access after user is created till verification is successful. This will be used for email, mobile number, facebook type of contact
--	--	--	--	--	--

4. Channel On-boarding

This transaction allows visiting user to onboard existing customer of the bank so that he can access his account through various online banking channels. This section describes the various configurations available for channel on-boarding.

Existing customer has to enter his existing account details which will be verified along with registered mobile number by one time password technique. However, parameters used for customer verification process are configurable. As per functionality bank can turn on/off validations for particular fields like email, first name etc. Bank can also configure type of user customer should be on-boarded to depends on type of customer. After verification customer will be navigated to channel selection screen where customer will be prompted to select channel groups as per his choice along with his choice of USER ID , which will be used later to login to FCDB application. Bank can configure login password functionality. If configured, password will be system generated or it can be as per user choice and same will be provided on channel selection screen. Once user and password entered are verified as per policy defined by bank, then user will be given access to selected channel groups and on-boarding process will be completed.

Channel Enrollment functionality has been given on login page with the hyper link. By selecting radio button with value “Yes, I am an existing customer of the bank”, customer can proceed with this process.

No	Property Name	Property Value	Day-0/Day-1	Table/File	Description
1	<id_entity>.COE.USERTYPE.I	Valid user type at FCDB end for e.g. EN1	Day 1	mstproperties	During on-boarding of customer who is non existing user, we can configure mapping between type of bank customer like individual, cooperate to user type maintained at FCDB end e.g. EN1 for retail , ECU for cooperate

2	<id_entity>.USERDEFINEDPWD.<usertype>	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding bank can ask user to enter password or not. If property is said to true then after customer verification then user will have to enter password by own one each for selected group. And if property is said to false then user don't have type enter password manually ,instead password will be system generated.
3	ACCTTYPE.CODE.<typeOfAccount>	Valid type account on host side e.g. CS, LF, TM	DAY 1	mstproperties	While customer verification step in channel on-boarding of customer. Based on relationship type selected by customer on GUI and based on this property type of account is set. And same is used while fetching data from host.
4	<id_entity>.CUSTOMER_DEBITCARD_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip debit card validation (debit card number and pin). If set to true, it will validate debit card details of saving account customer.
5	<id_entity>.CUSTOMER_EMAIL_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip email validation. If set to true, it will validate email of customer.
6	<id_entity>.CUSTOMER_FIRSTNAME_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will

	DATION_REQD				skip first name validation. If set to true, it will validate first name of customer.
7	<id_entity>.CUSTOMER_LASTNAME_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip last name validation. If set to true, it will validate last name of customer.
8	<id_entity>.CUSTOMER_DOB_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip date of birth validation. If set to true, it will validate date of birth of customer.
9	<id_entity>.CUSTOMER_NAMEONCARD_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip name on card validation. If set to true, it will validate name on card in case of credit card customer.
10	<id_entity>.CUSTOMER_EXPIRYDATE_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip expiry date validation. If set to true, it will validate name on card in case of expiry date of customer.
11	<id_entity>.CUSTOMER_CVV_VALIDATION_REQD	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will skip Card CVV validation. If set to true, it will validate name on card in case of expiry date of customer.
12	<id_entity>.CUSTOMER_OTP_VALIDATION	FALSE/TRUE	Day 1	mstproperties	Based on property, while channel on-boarding, it will

	_REQD				skip ONE TIME PASSWORD. If set to true, one time password is sent to registered mobile number customer
13	<id_entity>.COE.SKIP_CAPCTHA_VALIDATION	FALSE/TRUE	Day 1	mstproperties	Based on property, while on-boarding, it will skip image verification. If flag set to false, one verification image will be displayed to customer. He has to manually enter text shown in the image within expiry time of that password
14	<id_entity>.OTP_EXPIRY_TIME	Time in minutes	Day 1	mstproperties	While on-boarding if "<Entity>.CUSTOMER_OTP_VALIDATION_REQD" flag is set to true then, one time password is sent to registered mobile number of customer and is valid till specific time span. This property sets this time in minutes, after that OTP is treated as invalid , and customer has to re-initiate transaction or resend OTP
15	<id_entity>.DEFAULT_ACTIVATE.<usertype>	Y/N	Day 1	mstproperties	While on-boarding of customer whether user should be created through service request or not. If flag set to false then user is created without intervention of administrator. And if flag set to true then process is registered as service request and admin has to process the same using

					"Request processing" transaction
16	<id_entity>.ISSERVICEREQ.<user_type>	FALSE/TRUE	Day 1	mstproperties	While on-boarding of customer if Service Request flag is set to true then service request will be created using this dummy user id. As it is no login transaction and user id is not created for customer at this point, we need dummy user id to create service request.
17	<id_entity>.<user_type>.DUMMYUSERID	Valid FCDB user ID of that user type	Day 1	mstproperties	While on-boarding of customer if Service Request flag is set to true then service request will be created using this dummy user id. As it is no login transaction and user id is not created for customer at this point, we need dummy user id to create service request.
18	<id_entity>.DUMMYCUSTID	Valid customer ID	Day 1	mstproperties	While on-boarding of customer if Service Request flag is set to true then service request will be created using this dummy customer id.
19	CUSTOMERSOFTLOCKCNT	Maximum number of attempts customer can try entering his details for e.g. 3	Day 1	mstproperties	While customer verification step in channel on-boarding of customer. If customer enters any of the wrong details for n number of times then customer is locked for particular duration of time, called in

					"Soft lock" state. This property will be used for number of attempts customer can enter his details for verification before soft lock state. Lock can be released by batch process every fixed duration like daily, weekly. During those period customer will not be able to on-board customer.
20	<id_entity>.<typecontact>.COE.SKIP_CONTACT_VERIFICATION	FALSE/TRUE	Day 1	mstproperties	During on-boarding verification of particular contact type of customer like email, mobile number can be skipped. If flag is set to true for one contact type, that contact will not be verified during the process. And if flag is set to false or property is not present then customer has to verify that contact type before proceeding or he will be granted limited access after user is created till verification is successful. This will be used for email, mobile number, facebook type of contact

5. Forgot Password

This transaction will allow visiting user user can self Reset Password and unlock User account if It is in Unlocked or Normal Locked state. Forgot Password functionality is given for both types of users – Enrolled User and FCDB User.

Enrolled User (Registered User)

Registered User when uses Forgot Password Functionality, new password will be generated and sent to Registered Email ID.

FCDB User (Existing Customer)

All channel Users can avail this functionality, hence idtxn FPW is in DEFAULT_TXN list.

A number of Verification Steps will come up depending on configuration in ADTNL_PARAMS in mstchannelats.

User can be either allowed to Self Reset Password or Service Request will be generated to Admin for reset Password. This is indicated by ALLOW_RESET_PASSWORD property

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File	Description
1	DEFAULT_TXN	<Existing Default Txns>,FPW	Day-0	mstproperties	Available to all users by Default.
2	ADTNL_PARAMS	List of Verification steps or Name of Custom Handler.	Day-0	mstchannelats	E.g. MO,SQ, or Name of Custom Verification Handler Class
3	<Entity>. <UserType>. ALLOW_RESET_PASSWORD	TRUE/FALSE	Day-1	mstproperties	Describes whether to Allow self Reset Password (if 'TRUE'), else Service Request will be raised.

Secure Locking

New Type of Lock- Secure Lock is introduced.

A Secure Locked user account can be unlocked only by Admin.

Lock Reason is introduced using which Admin can specify custom Lock reason while locking any user.

Lock Types and Default Lock Reasons are maintained using following –

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File	Description
1	LOCK_TYPE	Y/S	DZ	apldata	Lock Types – Y - Normal Lock, S- Secure Lock
2	LOCK_REASON	LOCK_ADMIN	DZ	apldata	Secured locked by Administrator
3	LOCK_REASON	LOCK_FAIL	DZ	apldata	Normal lock due to wrong password entry multiple
4	LOCK_REASON	LOCK_SEQ	DZ	apldata	Secure Lock due to wrong answers entry for security questions multiple times.

6. Security Questions

Security Questions module has 3 parts-

1. Security Questions Maintenance by Admin
2. Set Security Question Answers by Channel User
3. Use Security Questions Answers for transaction Authentication

Following Properties and parameters are used -

Security Questions Maintenance –

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File	Description
1	DEFAULT_TXN	<Existing Default Txns>FCS,USQ	DZ	mstproperties	Added FCS and USQ - Security Questions Transactions - Available to all users by Default. FCS – Force Change Security Questions Answers USQ – Reset Security Questions Answers
2	TOTALSET	Integer	DZ	MSTENTITYUSERTYPES	Number of sets of Security Questions
3	QUESPERSET	Integer	DZ	MSTENTITYUSERTYPES	Number of Questions per set

Set Security Question Answers by Channel User

SR No	Property Name	Property Value	Day-0/Day-1	Table/File	Description
1	DEFAULT_TXN	FCS,USQ	DZ	mstproperties	Added FCS and USQ - Security Questions Transactions - Available to all users by Default.

					FCS – Force Change Security Quesitons Answers USQ – Reset Security Quesiton Answers
--	--	--	--	--	--

Security Questions Answers while transaction Authentication

SR No	Property Name	Property Value	Day-0/Day-1	Table/File	Description
1	SECLOCKCOUN	Integer	DZ	MSTENTITYUSER	Number of maximum allowed attempts before locking user.
2	NBRSECQUEST	Integer	DZ	MSTENTITYUSER	Number of security Questions to ask to user as part of transaction authentication
3	ISCASESENSITIVE	Y/N	DZ	MSTENTITYUSER	Indicates if Answers to Security Questions are case sensitive

7. Origination Non BPEL and BPEL and Origination Leads

This transaction allows Registered User as well as FCDB user to apply for Online Account Opening Process and Online Credit Card Application Process. This is possible through the below configuration into the application. Online Account Opening consists of two process i.e. BPEL flow and Non BPEL Flow.

Following are the steps to configure the Products for the Application which is going to be offered to the Registered User and FCDB User. This is applicable for Non BPEL and BPEL both.

Step 1:- Configure all the Product Categories in the table ORIG_PRODCATEG. This will be Entity and user type specific.

Considered Product Categories are as follows:-

Savings	-	S
Credit Card	-	Y
Loans	-	L
Current Accounts	-	C
Insurance	-	I

	ID_ENTITY	USERTYPE	PRODCATEGCODE
1	B001	EN1	C
2	B001	EN1	I
3	B001	EN1	L
4	B001	EN1	S
5	B001	EN1	Y

Step 2:- Configure all the Products belonging to each product categories configured in step 1 in the table ORIG_PRODUCTS. This is also entity, usertype and product category classified. Even Joint holders are also configured in this table

Considered Products are as follows:-

Savings

Regular Saving Account	-	RSA
Saving Account	-	SAV
Privilege Saving Account	-	PSA
Student Saving Account	-	SBA

Credit Cards

Credit Card Gold	-	CRC
------------------	---	-----

Loans

Business Loan	-	BLO
Home Loan	-	HLO
Personal Loan	-	PLO
Vehicle Loan	-	VLO

Current Accounts

Current Accounts Overdraft	-	CAO
Current Account Transactional	-	CAT

Insurance

Car Insurance	-	CAI
---------------	---	-----

Personal Accident and Health - PAI

ID_ENTITY	USERTYPE	PRODCATEGCODE	PRODCODE	JOINTHOLDERS
B001	EN1	C	CAT	
B001	EN1	C	CAO	
B001	EN1	I	PAI	
B001	EN1	I	CAI	
B001	EN1	L	PLO	
B001	EN1	L	VLO	
B001	EN1	L	BLO	
B001	EN1	L	HLO	
B001	EN1	S	SAV	2
B001	EN1	S	SBA	2
B001	EN1	S	PSA	2
B001	EN1	S	RSA	2
B001	EN1	Y	PCC	
B001	EN1	Y	SCC	
B001	EN1	Y	GCC	
B001	EN1	Y	CPD	
B001	EN1	Y	CRC	

The content for each of the product categories and their products are fetched from a content management server. Out-of-box, a file system based XML content management built in-house for OFCDB is available. To configure this content management system following configurations are required.

SR No	Property Name	Property Value	Day-0/Day-1	Table/File	Description
1	FCDB.PRODFILE S.LOC	Path for all the files are locates	Day-0	Mstproperties	From this location the Product category and product details will be fetched. File Name format for Product Category will be are as follows <Entity-Type>_<User-Type>_<lang>.xml File Name format for Product will be are as follows

					<Entity-Type>_<User-Type>_<ProdCode>_<lang>.xml
--	--	--	--	--	---

The product category and product descriptions are also maintained in APPLDATA table with DATANAME as ORIG_PROD_CATEG_NAME and ORIG_PROD_NAME.

If a content management system is available for providing the product category and product details, which includes short description, description, and display image, a new host adapter would have to be create such that it can interact with the content management system and return the required information. The host adapter responsible for fetching product content must return an object of the type `"com.iflex.fcat.services.apps.dtos.OrigProductResponseDTO"` where as the host adapter responsible for fetching product category content must return an object of the type `"com.iflex.fcat.services.apps.dtos.OrigProductCategoryResponseDTO"`. Below are the request IDs for which the adapters are to be configured.

For Product Category request ID – PRODCAT

For Product Details request ID – PRODUCT

Once this Configuration is done, design workflow for each product using the Workflow Configuration transaction from Bank Admin.

Out-of-box following screens are available for **Saving Accounts**:-

Applicant Details Step	:-	RRORG22
Linked Account Details Step	:-	RRORG23
Customize Account Step	:-	RRORG24
Upload Document Step	:-	RRORG31
Verify	:-	RRORG08 for Registered User RRORG131 for FCDB User

Confirm	:-	RRORG09 for Registered User RRORG132 for FCDB User
Penny Credit (Optional)	:-	RRPCV01
On-boarding	:-	RRCOE12

Out-of-box following screens are available for **Credit Cards**:-

Applicant Details Step	:-	RRORG41
Customize Account Step	:-	RRORG42
Upload Document Step	:-	RRORG31
Verify	:-	RRORG48 for Registered User RRORG141 for FCDB User
Confirm	:-	RRORG49 for Registered User RRORG142 for FCDB User

All of the above screens have been developed using Oracle FLEXCUBE Channel Workbench for Direct and Mobile Banking. Any new screens required should also be designed in Oracle FLEXCUBE Channel Workbench for Direct and Mobile Banking. The existing screens can also be modified using Oracle FLEXCUBE Channel Workbench for Direct and Mobile Banking as per requirements.

According to steps, the validation will be introduced into the system. Validation would always be specific to the product. So the request ID will be marked as isgenerictemplate = 'Y' which validate request ID specific validation in both table TXN_DATA_MASTER and TXN_DATA. Considered Validation Extensibility is as follows:-

Validation Extensibility	ISGenericTemplate	Y	Day-0	mstchannelats(Table)	As like Saving account for prospect in non login case Validations idRequest starts with RRORG08 for verify and
--------------------------	-------------------	---	-------	----------------------	---

					RRORG09 for confirm.
Validation Extensibility	ISGenericTemplate	Y	Day-0	mstchannelats(Table)	For FCDB user in login case idRequest starts with RRORG131 for verify and RRORG132 for confirm.

Product Features LOVs

In Origination we have introduced Product Specific LOV table i.e. ORIG_PRODFEATURESLOV. For Saving Account we have Cheque book type and Frequency of statement that has been maintained in the same table. This table holds LOV which specific to products. Their description is maintained in APPLDATA table with dataname as 'ORIG_CHQ_TYPE' and 'ORIG_FREQ_STMT'. In cheque book type '~' separator used for mentioning Type of Cheque Book and its Number of Leaves.

ID_ENTITY	USERTYPE	PRODCATEG	PRODCODE	LOVTYPE	CODE	SEQUENCE_NO
B001	EN1	S	SAV	CHQ_TYPE	M~10	
B001	EN1	S	SAV	CHQ_TYPE	M~25	
B001	EN1	S	SAV	CHQ_TYPE	PP~10	
B001	EN1	S	SAV	CHQ_TYPE	PP~25	
B001	EN1	S	SAV	CHQ_TYPE	P~10	
B001	EN1	S	SAV	CHQ_TYPE	P~25	
B001	EN1	S	SAV	FREQ_STMT	A	1
B001	EN1	S	SAV	FREQ_STMT	D	7
B001	EN1	S	SAV	FREQ_STMT	F	5
B001	EN1	S	SAV	FREQ_STMT	M	4
B001	EN1	S	SAV	FREQ_STMT	Q	3
B001	EN1	S	SAV	FREQ_STMT	S	2
B001	EN1	S	SAV	FREQ_STMT	W	6

Document Upload Section

Document Upload section consists of option group dropdown for Document Category and Document Type with mandatory document category need to be updated flag.

This is Entity, User type, Product Category, Product specific configurations. This consists of two table i.e. ORIG_DOCUMENTCONFIG and ORIG_DOCCATEGTYPEMAP.

ORIG_DOCUMENTCONFIG consists of document category mapping with reference to Entity, User type, Product Category, Product and its mandatory flag.

	ID_ENTITY	USERTYPE	PRODCATEG	PRODCODE	DOCCATEG	ISMANDATORY
1	B001	EN1	S	SAV	ADRPRF	Y
2	B001	EN1	S	SAV	EMPPRF	N
3	B001	EN1	S	SAV	IDENTPRF	Y
4	B001	EN1	S	SAV	LADPRF	N

Mandatory Flag defines which category document is to be mandatorily uploaded by user.

Document Category description is defined in APPLDATA table with dataname = 'ORIG_DOC_CATEG_DESC'.

ORIG_DOCCATEGTYPEMAP consists of the document types under the document category mapping with reference to Entity, User type, Product Category, Product.

	ID_ENTITY	USERTYPE	PRODCATEG	PRODCODE	DOCCATEG	DOCTYPE
1	B001	EN1	S	SAV	ADRPRF	DRILIC_ADR
2	B001	EN1	S	SAV	ADRPRF	ELEBIL
3	B001	EN1	S	SAV	ADRPRF	PASSPO_ADR
4	B001	EN1	S	SAV	ADRPRF	PHNBIL
5	B001	EN1	S	SAV	EMPPRF	EMPLET
6	B001	EN1	S	SAV	EMPPRF	SALDET
7	B001	EN1	S	SAV	IDENTPRF	DRILIC_IDT
8	B001	EN1	S	SAV	IDENTPRF	PANCAR
9	B001	EN1	S	SAV	IDENTPRF	PASSPO_IDT
10	B001	EN1	S	SAV	IDENTPRF	SSN
11	B001	EN1	S	SAV	LADPRF	BANSTM

After selecting the document category and type optgroup, User will upload documents the Repository system. The document will be uploaded into Oracle IPM tool. For this Oracle_IPM_SvcClient.jar need to be deployed, which will upload the files to the Oracle IPM. In this jar there is a property file which defines the authentication and IP of Oracle IPM where it's deployed. This Property files is under the Path com\ofss\util\OracleIPMAdapter.properties in the above Jar.

Origination Status

Status of the application is maintained in table ORIG_STATUS as day '0' with its access rights

	STATUS	STATUS_DESC	HOST_STATUS	MODIFY_ALLOWED	DATASTORE	APP_ACTOR	EXPDAYS
1	0	Initial Stage for Business User	...	BP	LOCAL	...	30
4	1	Saved	...	BP	LOCAL	...	15
14	2	Shared for Peer Review	...	BP	LOCAL	...	...
24	3	Request Completion	...	A	LOCAL	...	...
25	4	Expired	...	N	LOCAL	...	...
26	5	Cancelled	...	N	LOCAL	...	...
27	6	Submitted	...	N	LOCAL	...	...
32	7	Pending Authorization	...	N	LOCAL	...	...
5	8	Completed by us	...	BP	LOCAL	...	...
2	9	Under Process	UNDER_PROCESS	N	HOST	...	...
3	10	Application seen by peer	...	BP	LOCAL	...	...
6	11	Reviewed by peer	...	BP	LOCAL	...	...
7	12	New Account Opened	ACC_OPENED	N	LOCAL	...	30
8	13	Request Rejected	ORIG_REJECTED	N	LOCAL	...	10
9	14	Requires Corrections	MODIFY	BP	HOST	...	...
10	16	Requires Additional Documents	DOCS_REQUIRED	BP	HOST	...	...
11	17	KYC Verification in Process	...	N	LOCAL	...	...
12	18	KYC Successful	KYC_SUCCESS	N	LOCAL	...	...

Each Column Details is as follows:-

STATUS	This is the Status code of the application.
MODIFY_ALLOWED	This column specifies the modification access for any given user for applications in this status. B – Business User (Retail/Corporate) A – Administrator User P – Prospect (Enrolled User) N – Read-only
DATASTORE	This columns indicates the data store from which the application details are to be fetched. LOCAL – Fetch details from local tables HOST – Fetch details from the corresponding host system
APP_ACTOR	Indicates the capacity in which the current user can modify. O – Owner R – Peer Reviewer J – Joint holder/applicant A – Administrator user
EXPDAYS	Indicates the number of days an particular application can exist in the current status.

Status Description will be fetched from APPLDATA table for the

dataname = 'ORIG_STATUS_DESC'

The status flow are maintained in the database table ORIG_STATUS_FLOW. All the status and its respective next action records as of now is Day '0'.

	CODSTATUS	ACTION	ACTIONDESC	NEXTSTATUS	ACTOR	APP_ACTOR	ID_SUCC_MSG
30	0	SB	Submitted	6	PB		
29	0	SA	Save	1	PB		900528
39	1	SB	Submitted	6	PB		
40	1	SP	Shared for Peer Review	2	PB		
38	1	SA	Save	1	PB		900528
35	1	CL	Cancel	5	PB		900527
36	1	RB	Request Bank to Complete	3	PB		900529
59	2	SB	Submitted	6	PB	O	
58	2	SA	Save	2	PB	O	900528
19	2	RD	Reviewed	11	PB	R	900530
18	2	AS	Application seen by peer	10	PB		
55	2	CL	Cancel	5	PB	O	900527
57	2	RB	Request Bank to Complete	3	PB	O	900529
26	3	CU	Completed By Us	8	A		
27	3	SB	Submitted	6	A		

Each Column Details is as follows:-

CODSTATUS	Indicates the applications current status.
ACTOR	Indicates the action code available for the action performed on-screen.
NEXTSTATUS	Indicates the next status applicable for the CODSTATUS-ACTOR combination
ACTOR	Indicates the role type to which the current action is applicable.
APP_ACTOR	Indicates the actor type to which the current action is applicable.
ID_SUCC_MSG	Indicates the ID of the message to be displayed on-screen to the user on successful change in status.

Based on the status changes, some specific actions might be required to be performed. For performing such actions status handlers can be configured. The handlers are maintains in the database table **ORIG_STATUS_HANDLER**. This table maintains the service to be invoked once an applications status changes to the status indicated in the STATUS column of the table.

PRODCATEG	PRODCODE	STA	HANDLER	SERVICE	ID_ENTITY	PROCESSTYPE
1	S	SAV	6	com.iflex.fcatservices.apps.dtomappers.KYCCheckDTOMapper	OriginationApplicationServiceInterface.OriginationApplicationSei	B001
2	S	SAV	17	com.iflex.fcatservices.apps.dtomappers.KYCCheckDTOMapper	OriginationApplicationServiceInterface.OriginationApplicationSei	B001
3	S	SAV	18	com.iflex.fcatservices.apps.dtomappers.PCValidateDTOMapper	PennyCreditValidationServiceInterface.PennyCreditValidationSei	B001
4	S	SAV	24	com.iflex.fcatservices.apps.dtomappers.SubmitOrigAppDTOMa	OriginationApplicationServiceInterface.OriginationApplicationSei	B001

Each Column Details is as follows:-

PRODCATEG	Indicates the product category code for which the handler is set.
PRODCODE	Indicates the product code for which the handler is set.
STATUS	Indicates the status code for which the handler is set.
HANDLER	Indicates the DTO mapper to be invoked to fetch the request DTO for invoking the service specified in SERVIC column.
SERVICE	Indicates the service to be invoked
ID_ENTITY	Indicates the entity ID for which the maintenance is done.
PROCESSTYPE	Indicates whether the service is to be invoked online such that the calling

	service waits synchronously for a response from the service maintained in SERVICE column or whether the calling service returns without checking the response for the called service.
--	---

For New DTO Mapper

Make a Java file implementing `OriginDTOMapper` interface for method `doRequestMapping` which make a request DTO compatible to the service need to be called. Example is as follows.

```
public final class
    KYCCheckDTOMapper
implements
    OriginDTOMapper
{
    public RequestDTO doRequestMapping (
        TransactionContext      p_context
        , RequestDTO            p_dto
    ) throws Exception {
        //Request DTO Mapping Code
    }
}
```

For New Service Handler

This will be same as Normal Service class which we are using the existing product.

When application is submitted, on change of every status we check for handlers in this table. If handler is present we invoke corresponding service for further processing of application.

Online mode:

If handlers are configured as online, then the service is invoked synchronously and based on next status, further processing occurs (i.e as per 'Orig_Status_flow' table). Application will be

processed through all stages till it reaches final stage or will stop at stage which is marked as offline.

Offline mode:

Whenever any status is marked as offline, all applications in this status will be processed by timer (i.e OriginOfflineTimer.java) which runs on specific time interval. Before going into offline mode we save request as blob in 'ORIG_APP_OFFLINE_PROCESS' table.

Table **ORIG_APP_OFFLINE_PROCESS**:

Column name	Type	Description
IDSEQ	NUMBER	This column holds sequence id.
IDFCATREF	VARCHAR2(35)	This column holds Reference number of application.
REQUESTBLOB	BLOB	This column holds request as BLOB
SERVICENAME	VARCHAR2(255)	This columns hold name of service to be invoked.
SERVICEVERSION	NUMBER	This columns hold version of service to be invoked.

The reference number, service name and service version of the service which needs to be invoked from timer is stored in this table. This timer is configured in 'MSTTIMER' table. When timer executes at specified time, firstly it selects all rows from 'ORIG_APP_OFFLINE_PROCESS' table. It unmarshalls the request, then it invokes the related service for processing the application. Once it successfully processes the application then the row is deleted from the table.

Again whenever a particular stage is completed, we check handlers for that stage (i.e status). Depending on mode of handler above process is repeated.

Origination Application Sharing

Origination Application can be shared to joint holders/applicants as well as to Peers for review. Applications can be shared either via email or Facebook. For this Facebook app will be need to be plugged in.

For share application following properties is required to be configured.

SR No	Property Name	Property Value	Day - 0/D	Table/File	Description
-------	---------------	----------------	-----------	------------	-------------

			ay-1		
1	<ID_ENTITY>.ORG.EVENT_EXPIRY DURATION	Any Integer Value	Day-1	Mstproperties	This property specifies the no of days for which a shared application is valid. It also specifies the no of days for which the event will be created in case the application is shared via social media. Any integer value can be specified for this property.
2	<ID_ENTITY>.ORG.SHARE_CHANNELS	EM,FB	Day-1	Mstproperties	This property specifies the communication channels for sharing application. This is comma separated list, each specifying a communication channel. Description for each communication channel will be fetched from the table appldata with the combination of dataname "CONTACT_TYPE" and datavalue specified in the comma separated list for each item.
3	<ID_ENTITY>.ORG.SHARE_MANDATORY	Y	Day-1	Mstproperties	This property specifies if sharing of an application with co-applicants of the application is mandatory or not. Depending upon the value of this property (Y/N), the message shown to the user while applying for a product with one or more co-applicant varies.
4	<ID_ENTITY>.FB.APP.ID	Generated Id	Day-0	Mstproperties	An application must be created on Facebook side and generated id of that application must be set in this property

Proxy settings must be configured to access graph API of Facebook (these APIs are used for creating event on Facebook while sharing application)

Origination Application Tracking

Origination Application can be tracked from Admin User, FCDB User, Registered User. In Admin User

There are some properties used while tracking application are as follows

SR No	Property Name	Property Value	Day - 0/Day-1	Table/File	Description
1	<ID_ENTITY>.<USERTYPE>.ENG.NAME_FORMAT	{0} {3}, {1} {2}	Day-1	Mstproperties	This property describes the name format to be used, so that Admin can see the applicant's name in its own format as specified in Day Zero configuration. For Example : Applicant's name is displayed as : Mr Coelho Paul. Order is as follows : Salutation -> Lastname -> FirstName. Here {0} describes salutation . {1 } describes First Name {2} describes Middle Name {3} describe LastName.

Origination Download Application

File Download related configurations:

Added new content generator in B001.xml and T001.xml for pre login file download.

1. PLIPDF - for downloading PDF file
2. PLDWPDF - for print preview

SN o	Property Name	Property Value	Day- 0/Day-1	Table/File	Description
1	<ID_ENTITY>.<USERTYPE>.ORG.HOST.NAME	ID_HOST	Day 0	Mstproperties	Property to maintain Host name for origination.

Origination KYC ONLINE and OFFLINE

In STP application KYC status can be checked on either side depending on the configuration. If KYC need to be done on channel side, in that case ORIG_STATUS_HANDLER concept need to be introduced. As of now in the product for Non BPEL its configured in the same manner. For the same steps is as follows

After submit handler has been configured which decides whether need to post the application to the host or need to do KYC on channel side before submitting. Property for the same is follows.

S N o	Property Name	Property Value	Day- 0/Day- 1	Table/Fi le	Description
1	<ID_ENTITY>.ORIG_KYC_VERIFIER	ID_HOST FCDB/HOST	Day 0	Mstprop erties	Property identifies whether need to do KYC verification on FCDB side or not. FCDB value will decide the KYC Verification will be done on FCDB side. If the Value is HOST in that case the KYC will be done on Host.

If the above KYC property is for FCDB Side, we need to add one more handler (which is already present in the product) will decide the KYC will be done by OFFLINE mode or ONLINE mode. OFFLINE mode means FCDB Admin will do the KYC updates and in case of ONLINE KYC the Hook is provided through which the KYC API of the country or region will be consumed in result it will spontaneous KYC check. Property is as follows.

SN o	Property Name	Property Value	Day- 0/Day- 1	Table/Fi le	Description
1	<ID_ENTITY>.ORIG_KYC_VERIFY_TYPE	ONLINEKYC/OFFLINEKYC	Day 0	Mstprop erties	Property identifies KYC will be ONLINE mode or OFFLINE mode

If the Value is ONLINEKYC then the MSTHOSTINTERFACE table will be having the hook for the same idrequest. If any Bank need to configure the Online KYC then need to make their own adapter consuming their regions API for KYC on Idrequest ONLINEKYC and adapter should be compatible to give the response properly into FCDB Origination format.

Below records is the Status handler already enabled for Process handling.

For Saving Accounts

	PRODCATEG	PRODCODE	STATUS	HANDLER	SERVICE	ID_ENTITY	PROCESSTYPE
1	S	SAV	...	6 com.iflex.fcatservices.apps.dtomappers.KYCCheckDTOMapper...	OriginationApplicationServiceInterface.OriginationApplicationSei...	B001	ONLINE
2	S	SAV	...	17 com.iflex.fcatservices.apps.dtomappers.KYCCheckDTOMapper...	OriginationApplicationServiceInterface.OriginationApplicationSei...	B001	ONLINE
3	S	SAV	...	18 com.iflex.fcatservices.apps.dtomappers.PCVValidateDTOMapper...	PennyCreditValidationServiceInterface.PennyCreditValidationSei...	B001	ONLINE
4	S	SAV	...	24 com.iflex.fcatservices.apps.dtomappers.SubmitOrigAppDTOMa...	OriginationApplicationServiceInterface.OriginationApplicationSei...	B001	ONLINE

Origination Alerts

Alerts Configuration for Origination is as follows.

Sr. No.	Parameter Name	Paramter Type	Property Location	Priority	Detail usage & comments
1	APPLICANT	Functional	appldata	Day-0	The datavalue denotes all the possible fcdb status of application form for alerts to applicant. These status can be found in ORIG_STATUS table. For Near Expiry status we use - 1. DataValues ValueString ORIGSTATUS-1 ORIGNEAREXPIRY ORIGSTATUS1 ORIGAPPLSAVED ORIGSTATUS2 ORIGSHAREDFORPEER ORIGSTATUS3 ORIGREQCOMP ORIGSTATUS4 ORIGEXPIRED ORIGSTATUS5 ORIGCANCELLED ORIGSTATUS6 ORIGSUBMITTED

					ORIGSTATUS7 ORIGSAVEDBYADMIN ORIGSTATUS8 ORIGCOMPLETED ORIGSTATUS9 ORIGUNDERPROCESS ORIGSTATUS10 ORIGPEERACCESSED ORIGSTATUS11 ORIGPEERREVIEWED ORIGSTATUS12 ORIGACCOUNTOPENED ORIGSTATUS13 ORIGREJECTED ORIGSTATUS14 ORIGMAKECORRECTION ORIGSTATUS15 ORIGUNDERPROCESS ORIGSTATUS16 ORIGREQDOCS ORIGSTATUS18 ORIGKYCDONE ORIGSTATUS19 ORIGKYCFAILEDRETRY ORIGSTATUS20 ORIGKYCFAILED ORIGSTATUS21 ORIGPCVERIFY ORIGSTATUS22 ORIGPCVERIFYFAILEDRETRY ORIGSTATUS23
--	--	--	--	--	---

					ORIGPCVERIFYFAILED ORIGSTATUS24 ORIGPCVERIFYSUCCESS ORIGSTATUS25 ORIGACCEPTED ORIGSTATUS26 ORIGOFFERGENERATED ORIGSTATUS27 ORIGLOANAPPROVED
2	APPLICANTLEAD	Functional	apldata	Day-0	The datavalue denotes the fcdb status of leads application form for which the alert message should be different from normal alert messages. ORIGSTATUS30 - ORIGSUBMITTED ORIGSTATUS9 - ORIGUNDERPROCESSLEAD
3	PEER	Functional	apldata	Day-0	The datavalue denotes the fcdb status of application form for which the alert message should be sent to the peer. ORIGSTATUS2 - ORIGSHAREDDBYPEER
4	ORIGALERT	Functional	apldata	Day-0	Email subject, email body and sms message for origination alerts. The datavalue follows the naming convention as

					shown below: <u>Email Subject</u> <value strings for dataname=APPLICANT, APPLICANTLEAD and PEER>SUBJECT eg. Datavalue for email subject for KYC Done = ORIGKYCDONESUBJECT <u>Email Body</u> <value strings for dataname=APPLICANT, APPLICANTLEAD and PEER>BODY eg. Datavalue for email body for KYC Done = ORIGKYCDONEBODY <u>SMS</u> <value strings for dataname=APPLICANT, APPLICANTLEAD and PEER>SMS eg. Datavalue for sms for KYC Done = ORIGKYCDONESMS
--	--	--	--	--	---

7.1. BPEL Configuration for STP

Straight Through Process (STP) Savings Account Application can be submitted to UBS BPEL Gateway. This invokes Gateway Web Service.

Host Names

Host Name needs to be configured as BPEL for transactions as follows –

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File
1	<Entity>.<UserType>.<idRequest>.HOST.NAME	BPEL	DZ	Mstproperties

Above Property value needs to be maintained for Following idRequests:

1. RRORG09
2. RRORG132
3. RRORG06
4. RRORG10
5. RRORG49
6. GETRORIGAPP

BPEL Status codes maintenance

While sending complete Request to Gateway, FCDB needs to send Function name and Action Code in request. To complete particular stage what Function name and action code is needed are maintained using following property -

SR No	Property Name	Property Value	Day-0/ Day-1	Table/File
1	<BPEL STAGE NAME>.FN	BPEL	DZ	Function Name for completing this BPEL stage
2	<BPEL STAGE NAME>.AC	BPEL	DZ	Action code for completing this BPEL stage

Following are values for Function names and Action codes.

SR No	BPEL Stage Name	Function Name	Action Code	Description
1	MODIFYDET	STDSA060	PROCEED	Modify Application Details.
2	STOREDOC	STDSA070	PROCEED	Modify Application Details.
3	SAVEACCOUNT	STDSA074	PROCEED	Modify Application Details.

Products and Application related maintenance

Need following properties for supplying UBS BPEL Gateway supported values corresponding to FCDB values

SR No	Property Name	Property Value	Day-0/ Day-1	Table/File
1	APP.CAT. <ProductCode>	E.g. SAVIN, SAVHN, as maintained by HOST	DZ	Function Name for completing this BPEL stage
2	ACC.CLS. <ProductCode>	E.g. Savings Account, Current Account, as maintained by HOST	DZ	Action code for completing this BPEL stage
3	CHQBKTYPE.<Che queBookType>	Cheque Book Type as maintained by Host	DZ	Host Cheque Book Type corresponding to this ChequeBookType
4	OPNMODE.<Joint Mode>	Account Operation Modes as maintained by Host	DZ	Host Account Operation Mode corresponding to this Account Operation Mode

BPEL status flow

Application follows a status Flow in BPEL. At each stage, some Action can be performed on application, which moves application to another stage. A notification will also be received by FCDB when some Action is taken.

E.g. Application is in APRVACCOPEN stage. Marking it with Action- DOCMISSING will move it to STOREDOC stage. This will result in receipt of Notification with Status- APRVACCOPEN and Action – DOCMISSING.

FCDB will need to perform either or all of following operations on some of status change notifications.

Host Query

On receipt of some notifications, FCDB will need to Query the Application from Host to save some of Application details like BPEL Task Id, Penny Credit Contract Ref No., New Customer Number etc.

Which Status+Action combination notifications from Host needs Application Query is specified by following property

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File
1	<Entity>.<STATUS>.<ACTION>.HOST.QUERY.REQUIRED	Y/N	DZ	mstproperties

Following Status-Actions Pairs needs Host Query-

1. APRVACCOPEN.REJECT
2. NTFYCUST.PROCEED
3. INITPENCREDIT.NOTIFY

Application Status Change in FCDB

Some Application status changes in BPEL needs corresponding status changes in FCDB Via Notification provided by Core System(Applicable to Third Party as well). It is maintained in mstostappdatamap table as follows –

SR No	Table	IDTxn	Dataname	AppVal	HostVal
1	msthostapp datamap	ORG	ORIG_STATUS_MAPPING	FCDB Status code (It Will refer to hoststatus column of orig_status table)	Host STATUS OR Host STATUS.ACTION

Following Host status mappings are done

APPVAL	HOSTVAL
ACC_OPENED	NTFYCUST
DOCS_REQUIRED	APRVACCOPEN.DOCMISSING
KYC_SUCCESS	KYCDECISION.PROCEED
MODIFY	APRVACCOPEN.REJECT
VER_LINK_ACC	INITPENCREDIT

VER_LINK_ACC_FAIL	CANCELPENCRE
-------------------	--------------

Indicate Account Opening is Complete and Customer can be on-boarded
Particular Status + Action notification will indicate account Opening is complete.
This status action pair is maintained using following property.

SR No	Property Name	Property Value	Day-0 /Day-1	Table/File
1	<Entity>.ORG.FETCH.CUSTACC.DETAILS	<STATUS>.<ACTION> E.g.: NTFYCUST.PROCEED	DZ	mstproperties

7.2. Origination leads

Origination Leads configuration is as follows. This configuration will need to do from ADMIN user workflow configuration screen. Steps for Configuring any lead product are as follows.

1. Create product Category by inserting record into '[orig_prodcateg](#)' table for each entity and usertype.

E.g.: Suppose for Current Account Product with product Category 'C' entry in table would be:

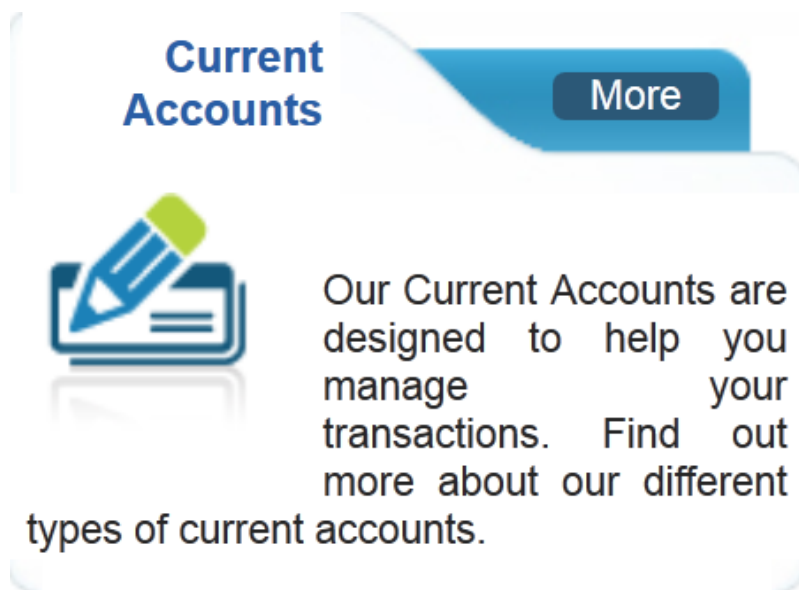
Entity (B001), User Type (EN1 – Retail User Type), Product Category(C).

Entity (B001), User Type (ECU – Corporate User Type), Product Category(C).

	ID_ENTITY	USERTYPE	PRODCATEGCODE
1	B001	ECU	C
2	B001	EN1	C

Table Description:

1. ID_ENTITY: Specify entity.
2. USERTYPE: Specify whether user is Retail User or Corporate User.
3. PRODCATEGCODE: Specify category of Product.



2. Corresponding to Product Category add product for that category in '[orig_products](#)' table for each entity and usertype .

E.g.: For Product Category configure above create Product as:

Entity (B001), User Type (EN1 – Retail User Type), Product Category(C), Product Code (CAO – Type of Current Account).

Entity (B001), User Type (EN1 – Retail User Type), Product Category(C), Product Code (CAT – Type of Current Account).

	ID_ENTITY	USERTYPE	PRODCATEGCODE	PRODCODE	JOINTHOLDERS
1	B001	EN1	C	CAO	
2	B001	EN1	C	CAT	

Table Description:

1. ID_ENTITY: Specify entity.
2. USERTYPE: Specify whether user is Retail User or Corporate User.
3. PRODCATEGCODE: Specify category of Product.
4. PRODCODE: Specify Product code corresponding to Product category.
5. JOINTHOLDERS: Specify whether joint holder is available for that product. For Leads it is not available .So always these column will be blank.

Current Accounts



Current Accounts Overdraft

A current account designed especially for you, keeping in mind high business demands and emergencies.

Features:

- Overdraft Facility
- Multi Location Fund Transfer
- Free Demand Drafts
- Free Cheque Collection
- Transfer of funds Higher daily withdrawal limit
- Zero Minimum Balance requirement

 [Apply Online](#)



Current Accounts Transactional

A current account that provides a host of services that caters to your every business need.

Features:

- Deposit and Withdrawal of Funds across international Locations
- Multi Location Fund Transfer
- Free Demand Drafts
- Free Cheque Collection
- Easy online account management
- Low Minimum Balance requirement

 [Apply Online](#)

- For each Product, create xml file in the format: Entity_UserType_ProductCategory_Language.xml which consists of details of Products available for that category along with features.
E.g.: B001_EN1_C_eng.xml.
- Create workflow configuration for Product through 'workflow configuration available in Admin', by creating Idrequest for viewing the product and for submission of product for each entity and usertype. For viewing the product step will be 1st and for submission step will be 2nd. In workflow maintenance Idrequest provided for 1st step will be visible in Leads screen as section 'Tell us your Requirement'. Rest 2 section i.e. Personal Information and Upload document section will be same for all the product. So only the product specific fields will be configured at 'Tell us your requirement section'. For all the different product Idrequest will be configured for the same. Even Validation for Leads will be product Specific, in that case the Confirm idrequest will be considered in mstchannelats as **Isgenerictemplate = 'Y'**. **TXN_DATA** and **TXN_DATA_MASTER** will be handles using confirm idrequest for validation.
E.g.: For Current Account – CAO Idrequest For Non Login shall be:
Viewing: RRORG51 and step sequence will be 1 and typeStep will be I (Initiate).
Submission: RRORG60 and step sequence will be 2 and typeStep will be C (Confirm).

	IDWORKFLOW	VERSION	IDSTEP	IDREQUEST	STEPNAME	STEPDESC	ADTNL_PARAMS	STEPSEQ	REPLICABLE	STATUS	TYPESTEP	IDCHANNEL
1	100	1	1	RRORG51	K_	K_		1			I	01
2	100	1	2	RRORG60	K_	K_		2			C	01

For Login Only Parameter 1 has to specified while creating workflow.

- For idrequest created through workflow configuration, create entry in 'mstchannelats' table.

	IDREQUEST	TYPPLUGIN	AUDITREQUIRED	IDCHANNEL	IDTXN	REQUIRESLOGIN	CONTENTSTYLE	NAMRESOURCE	IDSERVICE
1	RRORG51	C	Y	01	ORG	N	PLDHTML	genericwidgettemplate.xsl	currentAccountLead.xsl
2	RRORG60	C	Y	01	ORG	N	PLXML	getleadrefemcnumber.xsl	submitapplication.xsl

- For each product, configure 'document Type' to 'document Category' in 'orig_doccategtypemap' table for each entity, usertype, product category and product. Corresponding to each entity, usertype, product category, product and document Category configure whether document is mandatory or not in 'orig_documentconfig' table. For these Configuration details are available in Steps for Origination STP .
- Configure entry in 'orig_status_handler' table which is same as Origination STP.
- For submission of request, configure entry in 'msthostinterfac' table for each confirm Idrequest corresponding to product.

ID_ENTITY	IDHOST	IDREQUEST	VERSION	REQUEST_IFACE_MAPPER	RESPONSE_IFACE_MAPPER	INTERFACE_IMPL	ISENABLED
2	B001	UB11X	RRORG60	1		com.flex.feat.services.hostinterface.impl.FlexxmlHostAdapter	Y

E.g.: For CAO, confirm idrequest is RRORG60.

Current Accounts: For 'Tell us Your Requirement' Section typeStep will be 'I' and step Sequence will be '1'. For 'Submit Application' Section typeStep will be 'C' and step Sequence will be '2'. For Login Additional Parameter1 will be 'L'.

For Retail and Corporate Usertype: Out of the box following configuration are available.

Product Category	Product Code	Product	Idrequest for Non Login for 'Tell us Your Requirement' Section	Idrequest for Non Login for 'Submit Application' Section	Idrequest for Login for 'Tell us Your Requirement' Section	Idrequest for Login for 'Submit Application' Section
C	CAO	Current Account OverDraft	RRORG51	RRORG60	RRORG72	RRORG81
C	CAT	Current Account Transactional	RRORG52	RRORG61	RRORG73	RRORG82

Loans: For 'Tell us Your Requirement' Section typeStep will be 'I' and step Sequence will be '1'. For 'Submit Application' Section typeStep will be 'C' and step Sequence will be '2'. For 'Offer Generated Status' Section typeStep will be 'E' and step Sequence will be '3'. For Login Additional Parameter1 will be 'L'.

For Retail Usertype: Out of the box following configuration are available.

Product Category	Product Code	Product	Idrequest for Non Login for 'Tell us Your Requirement' Section	Idrequest for Non Login for 'Submit Application' Section	Idrequest for Login for 'Tell us Your Requirement' Section	Idrequest for Login for 'Submit Application' Section
L	PLO	Personal Loan	RRORG53	RRORG62	RRORG74	RRORG83
L	VLO	Vehicle Loan	RRORG54	RRORG63	RRORG75	RRORG85
L	HLO	Home Loan	RRORG55	RRORG64	RRORG76	RRORG84
L	BLO	Business Loan	RRORG56	RRORG65	RRORG77	RRORG86

For Corporate Usertype: Out of the box following configuration are available

Product Category	Product Code	Product	Idrequest for Non Login for 'Tell us Your Requirement' Section	Idrequest for Non Login for 'Submit Application' Section	Idrequest for Login for 'Tell us Your Requirement' Section	Idrequest for Login for 'Submit Application' Section
L	BLO	Business Loan	RRORG56	RRORG65	RRORG77	RRORG86

For Offer Generated Status: For all loans Products i.e. Personal Loan, Vehicle Loan, Home Loan and Business Loan Idrequest for Offer Generated Status is same as mentioned below:

Idrequest for Non Login for Getting Offer	Idrequest for Non Login for Accepting Offer	Idrequest for Login for Getting Offer	Idrequest for Login for Accepting Offer
RRORG126	RRORG123	RRORG126	RRORG127
RRORG126	RRORG123	RRORG126	RRORG127

Insurance: For 'Tell us Your Requirement' Section typeStep will be 'I' and step Sequence will be '1'. For 'Submit Application' Section typeStep will be 'C' and step Sequence will be '2'. For Login Additional Parameter1 will be 'L'.

For Retail Usertype: Out of the box following configuration are available.

Product Category	Product Code	Product	Idrequest for Non Login for 'Tell us Your Requirement' Section	Idrequest for Non Login for 'Submit Application' Section	Idrequest for Login for 'Tell us Your Requirement' Section	Idrequest for Login for 'Submit Application' Section
C	CAI	Car Insurance	RRORG58	RRORG67	RRORG79	RRORG88
C	PAI	Personal Accident and Health	RRORG59	RRORG68	RRORG80	RRORG89

Additional Document Required Status:

Idrequest for Non Login for Uploading Additional Document	Idrequest for Login for Modifying Application for Additional Document
RRORG122	RRORG128
RRORG122	RRORG128

Above All Product are configured. The feature is extensible for configuring new Leads product by configuring new Idrequest for 'Tell Us Your Requirement Section' and for 'Submitting Application' as mentioned above through 'workflow configuration available in Admin' for different entities and usertype. Follow all the steps mentioned above for configuring new Product.

8. Peer to Peer Payments

SN o	Module	Property Name	Property Value	Day- 0/Day-1	Table/File	Description
1	Peer To Peer Maintenance	PEER_BENE_IMAGE _ENABLED	Y or N	Day-1	mstproper ties	To enable/disable photo upload and display option.
2	Peer To Peer Maintenance	PEER_BENE_IMAGE _WIDTH		Day-1	mstproper ties	Image width(in pixels).
3	Peer To Peer Maintenance	PEER_BENE_IMAGE _HEIGHT		Day-1	mstproper ties	Image height(in pixels).
4	Peer To Peer Payments - Alerts	<ID_ENTITY>.BANK.L OGIN.URL		Day-1	mstproper ties	This property holds the login URL for the bank.
5	Peer To Peer Payments - Alerts	<ID_ENTITY>.P2P_R EG.CHASER		Day-0	mstproper ties	This property defines the frequency for chaser mails.
6	Peer To Peer Payments	<ID_ENTITY>.PPP.EV ENT_EXPIRYDURATI ON		Day-0	mstproper ties	This property defines the expiry duration for beneficiary registration.

7	Peer To Peer Payments	<ID_ENTITY>.EN1.CREDITGL.OR.HOLD	G or H	Day-0	mstproperties	This property decides whether the transfer amount will be credited to GL or will go on hold if the beneficiary is not registered.
8	Peer To Peer Payments	<ID_ENTITY>.<SOCIAL_MEDIA_ID>.CHANNEL.URL	//[host]:[port]/[APP]/channel.html	Day-0	fc.properties	URL for channel.html.
9	Peer To Peer Payments	<ID_ENTITY>.<SOCIAL_MEDIA_ID>.APP.ID	[APP_ID]	Day-0	fc.properties	Facebook App Id.
10	Peer To Peer Payments	<TXN_ID>.EVENT_NAME	PPP.EVENT_NAME - Funds Received	Day-0	apldata	The name of the event created while making payment to Facebook peer.
11	Peer To Peer Payments	<TXN_ID>.EVENT_DESC	PPP.EVENT_DESC - You have received funds from {0}. To claim the amount, login to {1}	Day-0	apldata	The description of the event created while making payment to Facebook peer.
12	Peer Bene Registration	<ID_ENTITY>.ORIG_ACTIVATION_EMAIL_EXPIRY_MINS		Day-0	mstproperties	Sets the email expiry duration.
13	Penny Credit Validation	<ID_ENTITY>.<USER_TYPE>.PCREQUIRE D	Y	Day-0	mstproperties	This property defines whether Penny credit is required or not. In DZ for ENC type

						user, set this flag as Y
14	Penny Credit Validation	PENNY_VALIDATION_MSG_ARGS_<ORIGIN>	2	Day-1	mstproperties	Property for storing varargs for penny success message
15	Penny Credit Validation	<ID_ENTITY>.PC.MAX.FORMATTED.AMOUNT		Day-0	mstproperties	Maximum amount for Penny credit Validation.
16	Penny Credit Validation	<ID_ENTITY>.PC.MAX.RETRY.COUNT		Day-0	mstproperties	Maximum retry allow for Penny validation

9. Verification Framework

This framework is used for adding configurable verification steps before invocation of any service. Once the verifications are complete the original service will be invoked. For example if there is a requirement as before confirmation of internal fund transfer user must have validate his E-mail id and also accept the term and conditions. You can configure these steps between verification and confirmation screens.

Configurations:

- Each verifier is identified by a verification code. The verification code and related verification request id mapping done in the **VERIFYREQUESTMAP** table.

TYPEVERIFICATION	IDREQUEST	COMMENTS
EM	RREMV01	Email Verification Type
CD	RRCDV01	Customer Details Verification Step
MO	RRMOV01	Mobile Verification Type
NP	RRFPW03	New Password - Self Reset Allowed Switch For Forgot Password
SQ	RRSQV01	Security Questions Verification Step - For Forgot Password
CO	RRCOE11	Mobile OTP verification step -Used in Channel Onboarding
AT	RRORG112	Used in Application Tracking For Non Login Case
TC	RRTNC04	Terms and Conditions Accept Step - Used in Originations
SM	RRSQV61	Security question authentication screen
NC	RRFPW63	New Password - Self Reset Allowed Switch For Forgot Password for Mobile channel
CM	RRCOE65	Mobile OTP verification step -Used in Channel Onboarding Usability

- The configuration steps which are required in the request is configured at **MSTCHANNELATS** table in **ADTNL_PARAMS** column against **VERIFICATION_PARAM** keyword. Here one can configure N number of steps in comma separated values.

IDREQUEST	RRFPW03	...
TYPPLUGIN	C	
AUDITREQUIRED	N	
IDCHANNEL	01	
IDTXN	FPW	
REQUIRESLOGIN	N	
CONTENTSTYLE	PLDHTML	...
NAMRESOURCE	genericscreentemplate.xsl	...
IDSERVICE	initiateforgotpasswordservice.xsl	...
IDAPP	RR	
FLAGPREPROCESS		
FLAGPOSTPROCESS		
NAMEOTRESOURCE	genericscreentemplate.xsl	...
NAMEOSRESOURCE	eos.xsl	...
ISONLYVALIDATEREQUEST	N	
IDSERVICE_EOT		...
IDSERVICE_EOS		...
IDREQUEST_EOT	RRFPW01	...
AUTHREQUIRED	N	
TXNPWDREQUIRED	N	
FLAGPAGINATION		
ADTNL_PARAMS	VERIFICATION_PARAM=MO,SQ,CD	...
FLGORCH	C	
ISONLYAUTHVALIDATE	N	
ISGENERICTEMPLATE	N	
HASEXTENDEDRESPONSE	N	
ALERTREQUIRED	N	
VALID_IDREQUEST		...
TOKEN1		...
TOKEN2		...
TOKEN3		...

- There is a configuration available for custom verification handler which can be plugged against the following priority
 - <ENTITY_ID>.<USER_TYPE>.<ID_REQUEST>.VERIFY_HANDLER
 - <ENTITY_ID>.<ID_REQUEST>.VERIFY_HANDLER

If any verification handler is plugged then that handler will invoked otherwise Default handler will invoked.

- Default Verify Handler is **com.iflex.fcat.services.apps.GenericVerifyHandler**. It is an abstract class. By default it return same number of verification steps which are maintain at **MSTCHANNELATS** table.
- Custom Verify Handler will extend **GenericVerifyHandler** class and override the **getApplicableSteps** method. This method returns comma separated verification steps.
 - For example in any transaction it is required that user's E-mail Id and Mobile number must verified before performing the transaction. So in **MSTCHANNELATS** table we configure both E-mail and Mobile verification steps and in Custom Verify Handler we evaluate the actual required verification step i.e. if user's E-mail is already verified so that step is not required.
- Every verification service has common signature as :

```

public final class
    EmailVerificationService

extends
    VerificationService

implements
    VerificationServiceInterface
{

}

```

Each verification service must implement ***VerificationServiceInterface*** and extends ***VerificationService***.

Service interface has following services:

```

public VerifierResponseDTO prepareValidate (
    TransactionContext      p_context
,    NullRequestDTO        p_request
) throws Exception;

```

This service provides the implementation of preparing call the business logic prior going into verification.

```

public VerifierResponseDTO initValidate (
    TransactionContext      p_context
,    NullRequestDTO        p_request
) throws Exception;

```

This service provides the implementation of initiating the business verification. This service does all prerequisites processing before calling the actual verification.

```
public VerifyTxnResponseDTO validate (  
    TransactionContext      p_context  
,    VerifierRequestDTO    p_request  
) throws Exception;
```

This service provides the implementation the business logic of validation.

```
public VerifyTxnResponseDTO skipVerification (  
    TransactionContext      p_context  
,    VerifierRequestDTO    p_request  
) throws Exception;
```

This service provides the implementation of method where verification can be skipped if configured as skip able and/or user wishes to skip verification step

- For adding new verification service these method should implemented except the **skipVerification** method. Since it is common across all verifier its implement in **VerificationService** itself.
- A step is skip able or not it is maintain at MSTPROPERTIES as prioritized as
 - <ENTITY_ID>.<USER_TYPE>.<ID_REQUEST>.SKIPABLE_STEPS
 - <ENTITY_ID>.<ID_REQUEST>.SKIPABLE_STEPS

This property has the comma separated values of skip able steps. By default all steps are mandatory.

- Commonly used methods are implemented as protected method in **VerificationService** and which are available to all verifier.
- Since actual verification logic is implemented in validate method. That service must have the custom extended response handler as **com.iflex.fcat.services.apps.VerifyExtResponseHandler** in **MSTSERVICES** table.
- For actual validation request id **FLAGPREPROCESS** value must be “8” and **HASEXTENDEDRESPONSE** value “Y”. And all verifier request have **IDTXN** as “VRS”

IDREQUEST	RRMOV02	...
TYPPLUGIN	C	
AUDITREQUIRED	N	
IDCHANNEL	01	
IDTXN	VRS	
REQUIRESLOGIN	N	
CONTENTSTYLE	PLDHTML	...
NAMRESOURCE	genericscreentemplate.xml	...
IDSERVICE	validate.xml	...
IDAPP	RR	
FLAGPREPROCESS	8	
FLAGPOSTPROCESS		
NAMEOTRESOURCE	genericscreentemplate.xml	...
NAMEOSRESOURCE	eos.xml	...
ISONLYVALIDATEREQUEST	N	
IDSERVICE_EOT		...
IDSERVICE_EOS		...
IDREQUEST_EOT	RRMOV01	...
AUTHREQUIRED	N	
TXNPWDREQUIRED	N	
FLAGPAGINATION		
ADTNL_PARAMS		...
FLGORCH	C	
ISONLYAUTHVALIDATE	N	
ISGENERICTEMPLATE	N	
HASEXTENDEDRESPONSE	Y	
ALERTREQUIRED	N	

- Verification can also be plugged after the service call. For calling verification after service call **extendedReturnCode** of **ResultDTO** will be set. Following value can set in **extendedReturnCode**
 - 10 - Verification is pending
 - 11 - Verification done successfully
 - 12 - Verification Required (after the service call)

These values are identifying the verification status.

- If any values are required to send in verification service when verification is plugged in service that can send in **udfFields** in the response DTO. **udfFields** whose name are started with “request.<fieldname>” are available in the channel request xml for next service call.

-
- When verification is call a reference number is generated against which the original request is stored. So it is required that that reference number must be pass in all verification screens. For that a hidden field ***fldtxnreferenceno*** must be available in all verification screens and its value is must be set. Its value can be set as set in following function in respective ***js*** file.