Release Notes
Oracle Financial Services Lending and Leasing
Release 14.10.0.0.0

September 2025

Part No. F35490-01

ORACLE
Financial Services

Table of Contents

1. RELEASE NOTES 1-1

1.1 BACKGROUND / ENVIRONMENT 1-1
1.2 PURPOSE 1-1
1.3 ABBREVIATIONS 1-1
1.4 PRODUCT SUMMARY 1-2
1.5 RELEASE HIGHLIGHTS 1-2
1.6 REVISION HISTORY 1-2

2. ENHANCEMENTS 2-1

2.1 CARES ACT- METRO II CHANGES 2-5
2.1.1 Overview 2-5
2.1.2 Description 2-5
2.1.3 Seed Data 2-9
2.2 LEASE TERMINATION ENHANCEMENTS 2-10
2.2.1 Overview 2-10
2.2.2 Description 2-10
2.2.3 Seed Data 2-15
2.3 AMOUNT DUE IN 'PROMISE TO PAY' 2-16
2.3.1 Overview 2-16
2.3.2 Description 2-16
2.3.3 Seed Data 2-18
2.4 AUTO CANCEL FUTURE PROMISES 2-19
2.4.1 Overview 2-19
2.4.2 Description 2-19
2.4.3 Seed Data 2-22
2.5 EXPORT TO EXCEL 2-23
2.5.1 Overview 2-23
2.5.2 Description 2-23
2.5.3 Seed Data 2-23
2.6 NEW LEASE LETTERS 2-24
2.6.1 Overview 2-24
2.6.2 Description 2-24
2.6.3 Seed Data 2-24
2.7 PAYMENT INTERFACE ENHANCEMENTS 2-25
2.7.1 Overview 2-25
2.7.2 Description 2-25
2.7.3 Seed Data 2-26
2.8 EVENT ENHANCEMENT (ADDED SECURITIZATION EVENT TYPE) 2-27
2.8.1 Overview 2-27
2.8.2 Description 2-27
2.8.3 Seed Data 2-27
2.9 REPORTING FUNCTION – DUE AMOUNT BETWEEN DUE DATES 2-28
2.9.1 Overview 2-28
2.9.2 Description 2-28
2.9.3 Seed Data 2-30
2.10 RESCISSION TRANSACTION ENHANCEMENT 2-31
2.10.1 Overview 2-31
2.10.2 Description 2-31
2.10.3 Seed Data 2-33
2.11 STATEMENTS ENHANCEMENTS 2-34
2.11.1 Overview 2-34
2.11.2 Description 2-34
2.11.3 Seed Data 2-37
2.12 MOCK STATEMENT ENHANCEMENTS 2-38
2.12.1 Overview 2-38

2.12.2 Description *2-38*
2.12.3 Seed Data *2-41*
2.13 PAYMENT REFUND REASON CODE 2-42
2.13.1 Overview *2-42*
2.13.2 Description *2-42*
2.13.3 Seed Data *2-45*
2.14 ASSET BILLING ENHANCEMENT 2-46
2.14.1 Overview *2-46*
2.14.2 Description *2-46*
2.14.3 Seed Data *2-46*
2.15 ACCOUNT UDF UPDATE SERVICE 2-47
2.15.1 Overview *2-47*
2.15.2 Seed Data *2-47*
2.16 SAME PRODUCER NUMBER ACROSS COMPANY 2-48
2.16.1 Overview *2-48*
2.16.2 Description *2-48*
2.16.3 Seed Data *2-50*
2.17 CHARGE OFF TRANSACTION ENHANCEMENT 2-51
2.17.1 Overview *2-51*
2.17.2 Description *2-51*
2.17.3 Seed Data *2-57*
2.18 NEW FEE CALCULATION METHOD (MASTER FEE) 2-58
2.18.1 Overview *2-58*
2.18.2 Description *2-58*
2.18.3 Seed Data *2-79*
2.19 WEB SERVICE ENHANCEMENTS 2-80
2.19.1 Overview *2-80*
2.19.2 Description *2-80*
2.19.3 Seed Data *2-80*
2.20 APPLICATION SEARCH WEB SERVICE - ENHANCEMENT 2-81
2.20.1 Overview *2-81*
2.20.2 Description *2-81*
2.20.3 Seed Data *2-81*
2.21 DECISION WEB SERVICE - GET 2-82
2.21.1 Overview *2-82*
2.21.2 Description *2-82*
2.21.3 Seed Data *2-82*
2.22 DECISION WEB SERVICE - PUT 2-83
2.22.1 Overview *2-83*
2.22.2 Description *2-83*
2.22.3 Seed Data *2-83*
2.23 ASSET FILE UPLOAD ENHANCEMENT 2-84
2.23.1 Overview *2-84*
2.23.2 Description *2-84*
2.23.3 Seed Data *2-84*
2.24 BULK GL FILE PROCESSING 2-85
2.24.1 Overview *2-85*
2.24.2 Description *2-85*
2.24.3 Seed Data *2-85*
2.25 MACHINE LEARNING FOR SERVICING QUEUE CREATION 2-86
2.25.1 Overview *2-86*
2.25.2 Description *2-86*
2.25.3 Seed Data *2-88*

3. APPENDIX: SEED DATA 3-89

4. PATCHES AND BUGS 4-1

5. SECURITY FIXES 5-1

6. LIMITATIONS AND OPEN ISSUES 6-1

7. COMPONENTS OF THE SOFTWARE 7-1

7.1 DOCUMENTS ACCOMPANYING THE SOFTWARE 7-1
7.2 SOFTWARE COMPONENTS 7-1

8. ANNEXURE – A: ENVIRONMENT DETAILS 8-1

9. ANNEXURE – B: THIRD PARTY SOFTWARE DETAILS 9-1

1. Release Notes

1.1 Background / Environment

Oracle Financial Services Software Limited has developed Oracle Financial Services Lending and Leasing solution. The suite is a comprehensive, end-to-end solution that supports full lifecycle of direct and indirect consumer/small business/business sector lending and leasing business operations with Origination, Servicing and Collections modules. This enables financial institutions to make faster lending, leasing, Renting and Subscription decisions, provide better customer service and minimize delinquency rates through a single integrated platform. It addresses each of the lending & leasing processes from design through execution. Its robust architecture and use of leading-edge industry standard technology components ensure almost limitless scalability.

1.2 Purpose

The purpose of this Release notes is to highlight the enhancements and bug fixes included in the Oracle Financial Services Lending and Leasing Release 14.10.0.0.0.

The release notes is also available in OTN. Refer to following link for any latest updates:
https://docs.oracle.com/cd/F35490_01/pdf/refdocs/ofsll_release_notes.pdf

1.3 Abbreviations

Abbreviation	Detailed Description
OFSLL	Oracle Financial Services Lending and Leasing
XML	Extensible Mark-up Language
XSD	XML Schema Definition
GL	General Ledger
XLF	Extended Log Format
JSF	Java Server Faces
EAR	Enterprise Application Archive
UI	User Interface
WSDL	Web Services Description Language
ACH	Automated Clearing House
MDB	Message Driven Bean
JMS	Java Messaging Service
IoT	Internet of Things

Abbreviation	Detailed Description
SME	Small and Medium Enterprise
UDF	User Defined Fields
FI	Financial Institute
TXN	Transaction (Monetary/Non-Monetary)
WS	websevrices

1.4 Product Summary

Oracle Financial Services Lending and Leasing Release built to meet various challenges faced by financial institutions. It addresses each of the lending and leasing processes from design through execution. Its unique value lies in its ability to provide the business with predefined processes and an excellent framework that takes care of business risk and compliance needs.

1.5 Release Highlights

The key highlight of this patch release is to fix the critical bugs and to enhance the industry specific requirements.

1.6 Revision History

Date	Name	Version	Change Reference
December 2020	OFSLL Team	1.0	Base release of 14.10.0.0.0
September 2025	OFSLL Team	1.1	Section 2.6, 2.7 and 28. Deprecated the Simple Account Creation, Account Dashboard and Document Chat bot Enhancements.

2. Enhancements

Enhancement	Description
Functional Enhancements	
CARES ACT - Metro II Changes	As per CARES Act, system needs to have the facility to report the account under FAQ 44 (Deferred payment) or FAQ 45 (Forbearance) or FAQ 58 (Natural Disaster), as per the guidance from CDIA. OFSLL is enhanced to report as per guidelines including K4 segment in the Metro II file (when account is deferment).
Lease Termination Txn Changes	Lease Termination and reversal transaction posting behaviour is standardized.
Promise to Pay -Due Amount recording	Promise to pay is enhanced to stamp the 'Due Amt' on the date of Promise.
Promise to Pay: All future promises are marked as broken automatically.	New facility is introduced to automatically cancel the future promises, if the current promise is recorded as 'Broken' and this behaviour is driven based on system parameter.
Export to Excel in selected screens	'Export to Excel' tool is provided in selective screen across the application.
New lease Letters	Introduced new lease letters → Payoff Quote and Paid in full letter
New Fee Calculation Method (Master Fee)	New 'Cycle Based' fee calculation methods (for Interest fee, Late fee and Collection Fee) are introduced for Master and individual accounts (Vacation Ownership specific industry).
Statement Enhancements	Currency, Product Name is added in Master Customer Statements, Customer Statements, and Master Customer Mock Statements.
Due Amount Calculation Functions between Due dates	New function for calculating Due Amount for the specific Due date and Account Number is provided.
Enable Charge-Off for Master Account	Facility is provided to post charge-off at the master account level and associated accounts based on the Transaction Parameter.
New Event Notification for pool code change	New Event trigger added to send Webhook notification on change of securitization pool change.
Customer/Master Based Payment Reversal Batch File Upload	Payment file upload enhanced to support reversal of payments for customer based payments or master based payments.
Customer/Master Based Payment Reversal Web Service	Payment Webservice supports reversal of payments for customer based payments or master based payments.

Enhancement	Description
Generic Payment Bulk upload Enhancements	Generic payment bulk upload file supports reversal of payments.
Same Producer Number across Company for all Branches	This enhancement facilitates to use a producer across all branches of a given company, irrespective of producer's branch, during account boarding and from UI.
Billing for Asset under Traded, Charged-Off and Voided.	These assets is not considered for posting charges using Rate table.
Innovations	
Intelligent Segmentation	Leveraged Oracle Database Machine Learning Capability to Identify the Clusters/Segments (Queues) that can be created based on the Account data. Setup user can select the Company, branch and Account condition and view the Machine Learning Algorithm suggestion and have capability to create Queue for selected Cluster from this new Screen. This screen developed based on Oracle JavaScript Extension Toolkit (JET).
Technical Enhancements	

Enhancement	Description
Technical Changes	• C to Java for Credit Bureaus (**Equifax & Experian**) OFSLL credit bureau processing such as request creation, bureau call, response parsing and response insertion for MDB flow was done using Pro*C code and for non MDB flow, except for bureau call, rest all processing was done by Pro*C code. As part of this enhancement we have moved the Pro*C code to JAVA code for Equifax & Experian bureaus also. **Equifax**: started supporting XML based request and response processing. The only supported Report is 'ACROFILE W/ON-LINE 02'. As part of response parsing, the header, score and human readable report information been shown on screen. **Experian**: started supporting JSON based request and response processing. The only supported Report is 'CREDIT PROFILE REPORT'. As part of response parsing, the header and score information been shown on screen. • Moved out the JSP /Java from DB → JSP dependencies has been deprecated for file processing in SaaS environment • File processing changes → Incoming & Outgoing File Uploads is supported for CLOB Y and N. • AQ processing changes → AQ-MDB flow is modified from processing the data from Advanced Queue table to process from Regular Database Table. Regular Table is introduced to de-queue the data from AQ table and save into this Table. MDB will process the data reading from the regular table. **Note the following** ➢ Customers who are migrating to current release from older release is recommended to refer the 'Upgrade Installation Guide' for configuration details. ➢ In future releases File based logging will be deprecated only LOB based support will be available ➢ OFSLL can be deployed on Oracle Autonomous Transaction Processing Dedicated (ATP-D). However, there are some limitations on the same. Refer to Database installation guide for more information.
Web Services Enhancements	
New Web service	New REST service(s) introduced: • Work Order Webservices – Post, Put and Get • Stipulations Webservice – Get • Setup: Pricing Webservice – Get • Decision Webservice – Get and Put • Simple Application Entry – Post • Update 'User Defined Fields' – Put

Enhancement	Description
Existing Web Services Enhancements	Updated following REST services • Application Search with Status/New Status • Added Payment Amount and Contract Details in Application Details Webservice **(GET - applicationDetails)** • Billing cycle, Company code, Balance paid and remaining outstanding balance for each balance type added to Account Details web-service **(GET – getAccountDetail)** • Payment Webservice supports reversal of payments for customer based payments or master based payments. • Post/PUT WS common Rectype→ Cleanup

2.1 CARES Act- Metro II Changes

2.1.1 Overview

CARES Act requires the facility to report the account under FAQ 44 (Deferred payment) or FAQ 45 (Forbearance) or FAQ 58 (Natural Disaster), as per the guidance from CDIA.

Act also, guides to report K4 segment in the metro II file (when account is deferment)

- If Portfolio Type = 'I' and 'C' – it is allowed
- Specialized Payment Indicator = 02
- Deferred Payment Start Date = Date, the first payment is due for deferred loans

2.1.2 Description

To report the K4 segment, user has to configure and system reports based on specific event as follows:

Post call activities with the following parameters

- Post the following call activities based on the Account Event Notification
- For **Forbearance**

Action	QR QUEUE/CONDITION REQUEST	QR
Result	OSPCC [OPEN OSPCC]	**Forbearance**
Condition	NONE – if account doesn't have an active SPCC condition SPCC – If account has active SPCC condition	NONE
Reason	CP	Null

- For **Natural Disaster**

Action	QR QUEUE/CONDITION REQUEST
Result	OSPCC [OPEN OSPCC]
Condition	NONE – if account doesn't have an active SPCC condition SPCC – If account has active SPCC condition
Reason	AW

- For **Deferred Payment**

Action	QR QUEUE/CONDITION REQUEST	QR
Result	DP	**Deferred Payment**
Condition	NONE	NONE

Reason	<Null>	Null

Affected by natural or declared disaster [FAQ 58]

Reports account with Special Comment Code “AW” (Affected by natural or declared disaster).

If the Account has **Natural Disaster** & **Differed Payment** condition and SPCC code is reporting as ‘AW’

Metro II Fields	**Metro ii File Data**
Terms Duration	Blank
Terms Frequency	D
Scheduled Monthly Payment Amount	Zero
Account Status Code	11
Amount Past Due	Zero
Special Comment Code	AW
Payment History Profile	Use Character D for the months where payments are deferred.
K4 segment	Introduce and report following Specialized Payment Indicator = 02 Deferred Payment Start Date = Special Category Start date in MMDDYYYY Balloon Payment Due Date = Null Balloon Payment Amount = 0

Note: if account is having only Natural Disaster / AW, then it reports as existing.

Account in forbearance [FAQ 45]

If the account has CP Special comment code and marked as ‘Account in Special Category’ then system Reports account with Special Comment Code “CP” (Account in forbearance)

If the Account has SPCC condition and SPCC code is reported as ‘CP’

Terms Duration	Blank→ if ‘Scheduled Monthly Pmt is zero’ else actual account term
Terms Frequency	D →if ‘Scheduled Monthly Pmt is zero’ else actual account term

Payment History Profile	Increment the Payment History Profile with value D if 'Scheduled Monthly Pmt is zero'
Special Comment Code	CP
K4 segment	Introduced and reports as follows: Specialized Payment Indicator = 02 Deferred Payment Start Date = Forbearance condition Start date in MMDDYYYY Balloon Payment Due Date = Null Balloon Payment Amount = 0

Account in Deferral Payment [FAQ 44]

If the account marked as 'Account in Special Category' and having 'Deferred Payment' condition

Reports the following fields as

Terms Duration	Blank
Terms Frequency	D
Scheduled Monthly Payment Amount	0
Account Status	11
Payment History Profile	Increment the Payment History Profile with value D
Amount Past Due	Zero
K4 segment	Introduced to report following K4 segment Specialized Payment Indicator = 02 Deferred Payment Start Date = Deferred Payment condition Start date in MMDDYYYY Balloon Payment Due Date = Null Balloon Payment Amount = 0

Reporting Guide Lines

Account Condition	SPCC Code	Guidelines
Natural Disaster	AW	No change
Natural Disaster + Deferral Payment	AW	FAQ 58
Forbearance	CP	FAQ 45
Forbearance + Deferral Payment		

Account Condition	SPCC Code	Guidelines
Deferral Payment	NA	FAQ 44

If both Natural Disaster and Forbearance are posted on the account, go with latest SPCC code available on the account.

Changes in Metro ii file creation

Following fields would be reported to the Metro II File under K4 Segment

- Specialized Payment Indicator
- Deferred Payment Start Date
- Balloon Payment Due Date
- Balloon Payment Amount
  - If the account is under Special Category, then while executing the SET-ODD2 > CBUUTL_BJ_100_02 batch job, system creates the above four new fields data as per the stated guild lines in above tables
  - While executing the SET > ODD > CBUUTL_BJ_100_03, based on the value configured under CBU_FILE_FORMAT company parameter, system generates the file as follows

CBU_FILE_FORMAT parameter Value	Header Record begin with		Base Segment + J2 segments + L1 + K4 segments to be include in file?						Trailer Record
		426 Base Segment begin with	J2_1	J2_2	J2_3	J2_4	K4	L1	
710	710	710	Yes	No	No	No	Yes	Yes	710
910	910	910	Yes	Yes	No	No	Yes	Yes	910
1110	1110	1110	Yes	Yes	Yes	No	Yes	Yes	1110
1310	1310	1310	Yes	Yes	Yes	Yes	Yes	Yes	1310

- If system creates the file with K4 segment, i.e., user would have configured one of the above values 710, 910, 1110, or 1310 then system incorporates K4 segment as before L1 segment

Field	Field Name	Required	Length	Position	Value
1	Segment Identifier	Y	2	1-2	constant value K4
2	Specialized Payment Indicator	Y	2	3-4	as per the stated guild lines in above tables
3	Deferred Payment Start Date	A	8	5-12	

Field	Field Name	Required	Length	Position	Value
4	Balloon Payment Due Date	A	8	13-20	
5	Balloon Payment Amount	A	9	21-29	
6	Reserved		1	30	Blank fill

- In the Trailer Record (T1), system updates the count of total records of such special category base record count to be updated in the position 36 [309-317].

2.1.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.2 Lease Termination Enhancements

2.2.1 Overview

Lease Termination posting and reversal related enhancements are handled and standardized the behaviour and process.

2.2.2 Description

Applicability of changes:

- Termination transaction posting
- Changes in Balances Screens
- Payoff quote

Termination Posting Impact - Summary

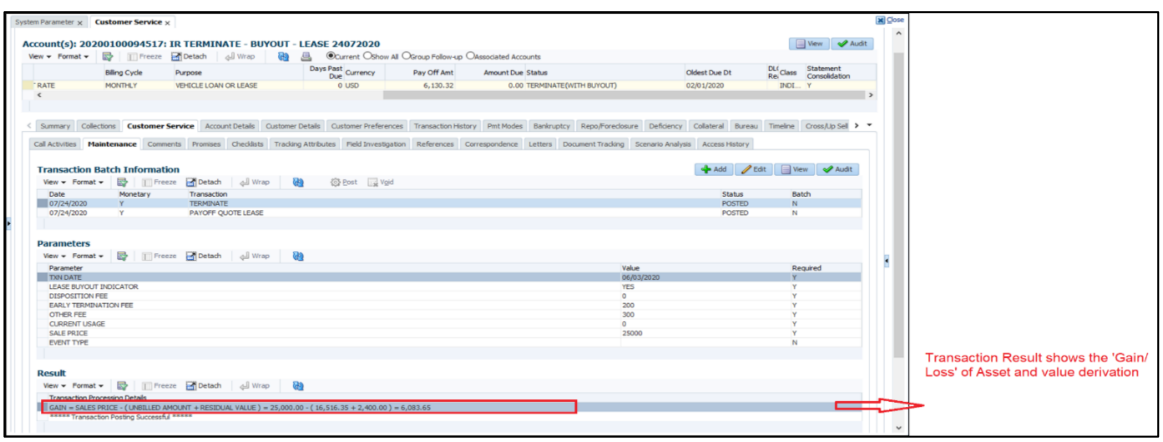
	Rent Factor – without Buyout	Rent Factor – with Buyout	Interest Rate – without Buyout	Interest Rate – with Buyout	Comment
Early Termination (Before Maturity Date)	Rent rebate posted – If applicable	Rent rebate posted – If applicable	'Interest Accrual' transaction will be posted	'Interest Accrual' transaction will be posted	
Normal Termination (Post Maturity Date)	--	--	'Interest Accrual' transaction will be posted	'Interest Accrual' transaction will be posted	User may have to post payment transaction multiple times; as 'Interest accrual' happens after termination and no payoff quote is available to post exact payment. Residual value is included in final statement.
Due Date History	Payment Received flag gets updated to Y (Irrespective payment received or not)	Payment Received flag gets updated to Y (Irrespective payment received or not)	Payment Received flag gets updated to Y (Irrespective payment received or not)	Payment Received flag gets updated to Y (Irrespective payment received or not)	If Termination reversed, this flag reinstates to normal.

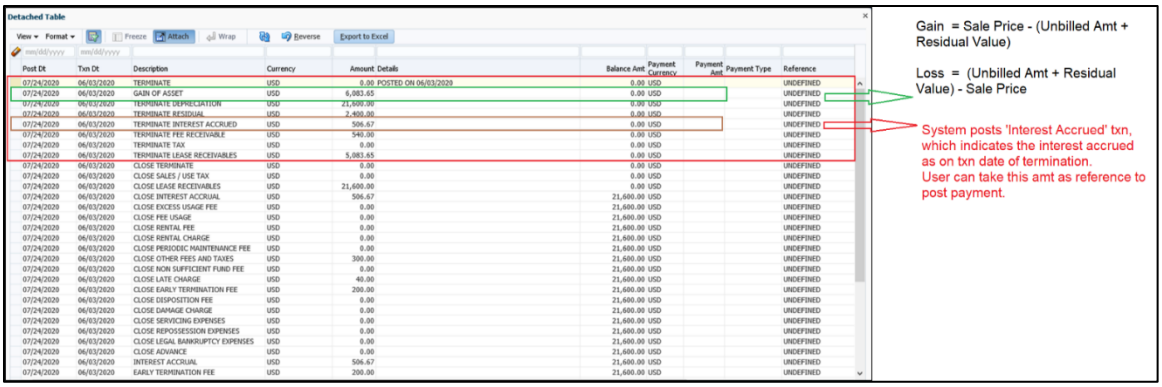
Balances (If Termination posted)	'Current Balances' on account would be moved to 'Terminate' Balances. While moving the balances, system 'Terminates' all individual balances and any outstanding dues (Lease Receivables, Rent, Fee etc) would be tracked against the consolidated balance called 'Termination Balance'.	'Current Balances' on account would be moved to 'Terminate' Balances. While moving the balances, system 'Terminates' all individual balances and any outstanding dues (Lease Receivables, Rent, Fee etc) would be tracked against the consolidated balance called 'Termination Balance'.	'Current Balances' on account would be moved to 'Terminate' Balances. While moving the balances, system 'Terminates' all individual balances and any outstanding dues (Lease Receivables, Rent, Fee etc) would be tracked against the consolidated balance called 'Termination Balance'.	'Current Balances' on account would be moved to 'Terminate' Balances. While moving the balances, system 'Terminates' all individual balances and any outstanding dues (Lease Receivables, Rent, Fee etc) would be tracked against the consolidated balance called 'Termination Balance'.	
Transactions	All balances are terminated (negated) and specifically 'Inventory' transaction is posted on account. Inventory = Unbilled Amt + Residual Value	All balances are terminated (negated) and **Following txns posted:** Gain /Loss of Asset Terminate Depreciation Terminate Residual Terminate Interest Accrued Terminate Fee Receivable Terminate Tax Terminate Lease Receivables ------------------------------Gain / Loss on Asset = Sale Price - (Unbilled Amt + Residual Value)	All balances are terminated (negated) and specifically 'Inventory' transaction is posted on account. Inventory = Unbilled Amt + Residual Value	All balances are terminated (negated) and **Following txns posted:** Gain /Loss of Asset Terminate Depreciation Terminate Residual -------------------------------- Gain / Loss on Asset = Sale Price - (Unbilled Amt + Residual Value)	Gain / Loss and Inventory information is shown as 'Termination Txn'→ **Result**. These are shown under 'Transaction History' and **not shown** under Balances.

Balances → Current Balance	Inventory balance is not accounted in totaling of balances in Lease → Customer Service → Transaction History → Balances → Current Balance. (Label also read as Lease Current Balance Total (Excluding Inventory))		Inventory balance is not accounted in totaling of balances in Lease → Customer Service → Transaction History → Balances → Current Balance. (Label also read as Lease Current Balance Total (Excluding Inventory))		
Payoff Quote	Not generated after termination	Not generated after termination	Not generated after termination	Not generated after termination	
Statement	Not generated after termination	Not generated after termination	Not generated after termination	Not generated after termination	
Paid off (after termination)	Pay amount in full and any excess amount can be refunded.	Pay amount in full and any excess amount can be refunded.	Pay amount in full and any excess amount can be refunded.	Pay amount in full and any excess amount can be refunded.	
Charge off (after termination)	Only allowed to post after termination. All balances would be moved to 'Deficiency' bucket. Any payment after 'charge off' would show under 'Recovery' bucket. Account will be closed after certain no. of days (based on system parameter)	Only allowed to post after termination. All balances would be moved to 'Deficiency' bucket. Any payment after 'charge off' would show under 'Recovery' bucket. Account will be closed after certain no. of days (based on system parameter)	Only allowed to post after termination. All balances would be moved to 'Deficiency' bucket. Any payment after 'charge off' would show under 'Recovery' bucket. Account will be closed after certain no. of days (based on system parameter)	Only allowed to post after termination. All balances would be moved to 'Deficiency' bucket. Any payment after 'charge off' would show under 'Recovery' bucket. Account will be closed after certain no. of days (based on system parameter)	

Restriction on transactions after posting termination (System doesn't allow to post these transactions, once termination is posted)	**Monetary txns**: LEASE CHANGE PAYMENT AMOUNT RESCHEDULE BILL CYCLE LEASE EXTENSION EVERGREEN LEASE **Non-monetary txn:** ACCOUNT CONTRACT MAINTENANCE ACCOUNT PAYMENT MODE MAINTAINANCE STATEMENT PREFERENCE MODE ADD NEW ASSET and ACCOUNT EVENT NOTIFICATION (Involuntary repossession)"	**Monetary txns**: LEASE CHANGE PAYMENT AMOUNT RESCHEDULE BILL CYCLE LEASE EXTENSION EVERGREEN LEASE **Non-monetary txn:** ACCOUNT CONTRACT MAINTENANCE ACCOUNT PAYMENT MODE MAINTAINANCE STATEMENT PREFERENCE MODE ADD NEW ASSET and ACCOUNT EVENT NOTIFICATION (Involuntary repossession)"	**Monetary txns**: LEASE CHANGE PAYMENT AMOUNT RESCHEDULE BILL CYCLE LEASE EXTENSION EVERGREEN LEASE **Non-monetary txn:** ACCOUNT CONTRACT MAINTENANCE ACCOUNT PAYMENT MODE MAINTAINANCE STATEMENT PREFERENCE MODE ADD NEW ASSET and ACCOUNT EVENT NOTIFICATION (Involuntary repossession)"	**Monetary txns**: LEASE CHANGE PAYMENT AMOUNT RESCHEDULE BILL CYCLE LEASE EXTENSION EVERGREEN LEASE **Non-monetary txn:** ACCOUNT CONTRACT MAINTENANCE ACCOUNT PAYMENT MODE MAINTAINANCE STATEMENT PREFERENCE MODE ADD NEW ASSET and ACCOUNT EVENT NOTIFICATION (Involuntary repossession)"	

Gain / Loss Transaction Posting:

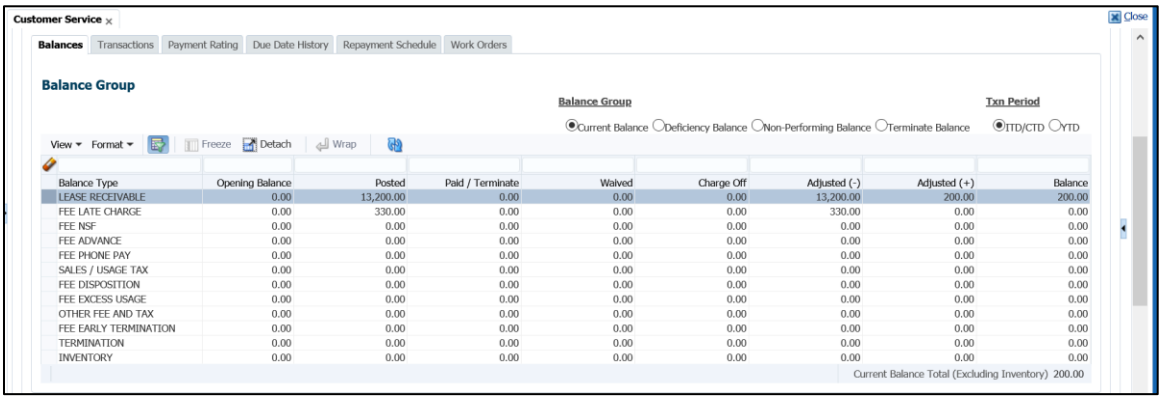




Changes in Servicing → Transaction History → Balances

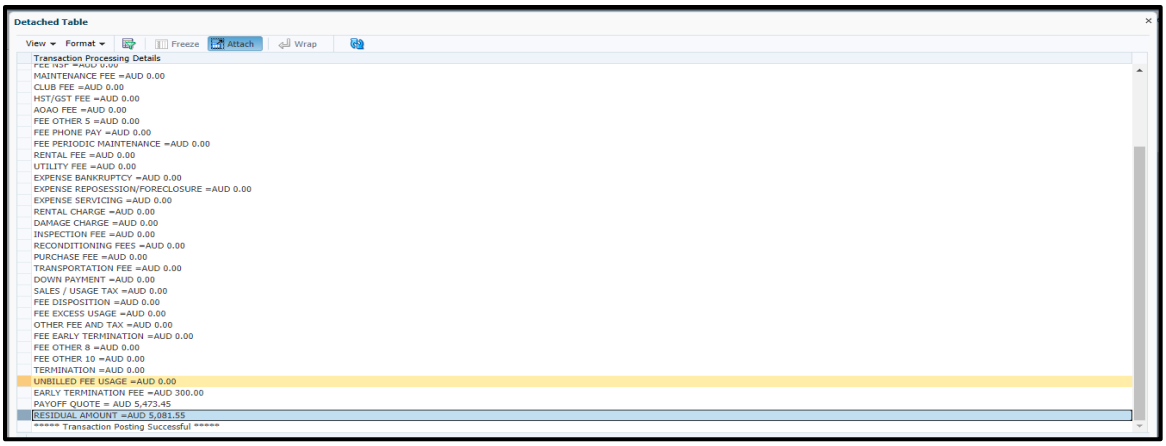
As part of this exercise, following changes were taken up in the balances screen: (Made applicable for Loan, Lease & Line).

- Terminate Balance Radio Button
  - Added columns → Posted, Balance
  - Removed columns → Terminate
- Lease → Current Balances Radio Button
  - Footer totals label renamed as 'Lease Current Balance Total (Excluding Inventory)
  - Total value always excludes the 'Inventory' balances (as this amount is not tracked as 'Dues' in OFSLL and posted for accounting purpose only.



Changes in 'Payoff Quote'

'Residual Value' will be shown as separate line item in 'Payoff Quote' and would not be part of 'Payoff Amount'.



2.2.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.3 Amount Due in 'Promise to Pay'

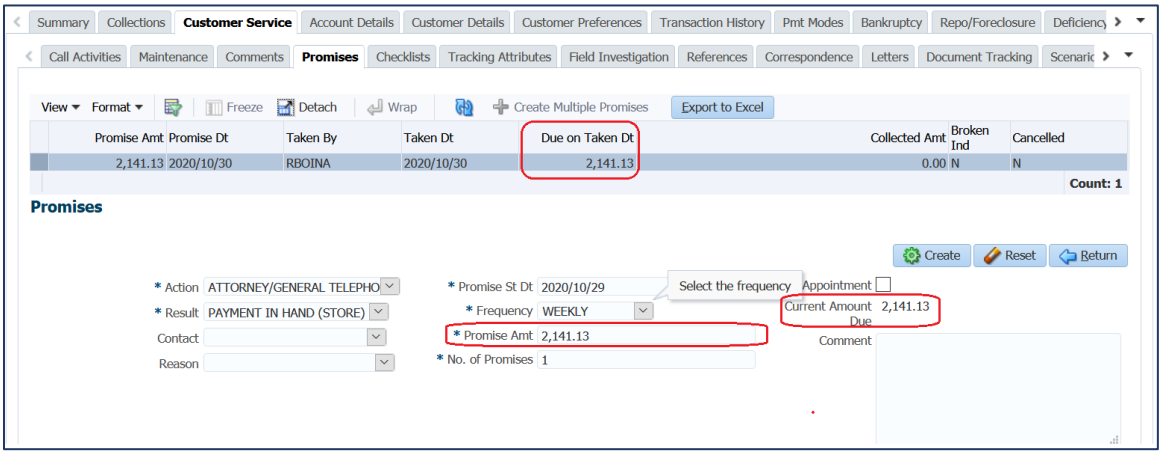
2.3.1 Overview

Currently, system doesn't store the history of amount past due on daily basis. But while taking the Promises, user wants to capture the Amount Due i.e., user would take the promise during the day but amount due is be shown on that system process date. This is for a history and audit purpose and evaluate the collector performance.

2.3.2 Description

Changes in Customer Service > Customer Service > Promises > Multiple Promise to Pay Screen

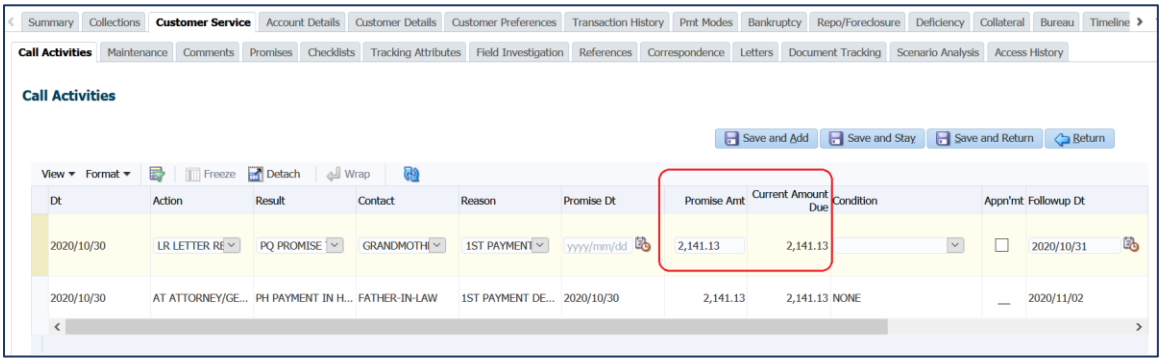
- Added a new read-only field '**Current Amount Due'** in add screen, after Appointment check box; In the header, it will be showed it as 'Due on Taken Dt'
- Once, user click on 'Create Multiple Promise' button, system shows the Current Amount Due with accounts' Amount Due
- Once user selects the action, and result, system also defaults the Promise Amt with accounts' Amount Due
- After create and submitting the record, system stamps that Amount Due in the Call Activities and Promises tabs



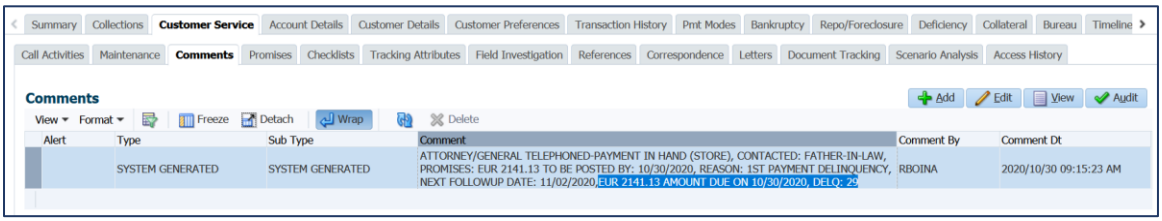
Changes in Customer Service > Customer Service > Call Activities

- While creating the record, added a new read- only field 'Current Amount Due' in this screen after Promise Amount; After saving the record, it will be showed as 'Due on Taken Dt'.
- While adding the record, system will show the amount due of that account in 'Due on Taken Dt' field and on saving the record, system will store this amount on that call activity
- After selection of Action and Result, system will default the Promise Amt with accounts' amount due.

Note: In both the screens, system uses the Customer Service > Customer Service > Accounts > **Amount Due** Field value while creating the above records for all product types (loan/line/lease)



Also, after saving the record system will posts a comment on to the account, in that, system will append the Amount Due as '<Account Currency Code> <Values of Amount Due> Amount Due on <GL Date>



Customer Service / Collections > Right Splitter > Call Activity

- If user using the right splitter to post the call activity, then also system will stamp the amount due in the call activity and then create the comment with amount due

Non-monetary Txn changes

- When user post the 'Account Event Notification' transaction, if the system creates the call activities automatically then also system defaults the Amount Due then create the comment with amount due

File upload changes [SET-IFP > ICAPRC_BJ_100_01 > OFFLINE CALL ACTIVITY POSTING]

- When user upload the call activity file, system will default the amount due and create the comment with amount due

Web service changes

- When user post the call activity, system will stamp the amount due and then create the comment with amount due
- When user post the Account Event Notification transaction through Generic Post Transaction, then system will stamp the amount due and then create the comment with amount due

Archive and Purging – System will do Archive/Purge of this new field after running the below batch jobs

- PACARC_BJ_100_01 - ARCHIVE ACCOUNT DATA TO OTABLES
- PACARC_BJ_100_02 - ARCHIVE ACCOUNT DATA TO OOTABLES
- PAPARC_BJ_100_01 - ARCHIVE APPLICATION DATA TO OTABLES
- PAPARC_BJ_100_02 - ARCHIVE APPLICATION DATA TO OOTABLES
- PJRPAP_BJ_100_01 - PURGE APPLICATION DATA
- PJRPAC_BJ_100_01 - PURGE ACCOUNTS DATA

Call activity through Events

- If system creates the call activity through an event, then system will default the amount due and create the comment with amount due

2.3.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.4 Auto Cancel Future Promises

2.4.1 Overview

OFSLL system currently supports Promise to Pay feature capturing along with the future pay date, amount and other information captured via the 'Add Call Activity' section. (Right side collapse / restore pane). It also supports the 'Multiple Promises Option' via which the multiple promise dates can be captured along with the other promise details.

Some of financial institution have practice of cancelling all the future promises, once any of the promise is not fulfilled by the customer. OFSLL is enhanced to handle this business use case to automatically cancel/broken the future promises, once the promise is broken.

2.4.2 Description

Introduced a new lookup code 'FUTURE_PROMISE_MTHD_CD: FUTURE PROMISE METHOD CODE'

Lookup code	Lookup Description	Sort	Enable	System Defined
UNDEFINED	No Action on Future Promises	0	Y	Y
BROKEN	Mark Current and Future Promises as Broken	1	Y	Y
CANCEL	Mark Current as Broken but Future Promises as Cancelled	2	Y	Y

Introduced a new record in SET_SYP_COMPANY as follows

- Description: FUTURE PROMISE HANDLING METHOD
- Enabled: No
- LOV Type: LOOKUP
- LOV Validation Ind: Y
- Lookup Type: FUTURE_PROMISE_MTHD_CD
- Default Value: UNDEFINED

Introduced new company parameter 'FUTURE PROMISE HANDLING METHOD' with following above lookup reference.

Default option: UNDEFINED. If the parameter is disable and not configured for specific company, then system will treat it as UNDEFINED.

System behaviour:

- UNDEFINED - current behaviour. No impact on future promises.
- BROKEN – Marks current as well as all future active promises as broken.
- CANCEL – Marks current promise as broken and all future promises as Cancelled.

Example 1:

GL date is June 1, 2020

Promise Amount: 100

No payment transactions on June 1,2020

EOD, when the SET-COL CPPPRC_BJ_100_01 batch job gets executed

Parameter Value		BROKEN		
Taken Date	Promise Date	Broken Ind	Cancelled Ind	System action
May 1, 2020	June 1, 2020	Y	N	
May 1, 2020	June 5, 2020	Y	N	Marked 6/5/2020, 6/15/2020, 6/20/2020, 6/25/2020 Promise(s) as broken by system
May 10, 2020	June 15, 2020	Y	N	
May 20, 2020	June 20, 2020	Y	N	
May 20, 2020	June 25, 2020	Y	N	

Parameter Value		CANCEL		
Taken Date	Promise Date	Broken Ind	Cancelled Ind	**System action**
May 1, 2020	June 1, 2020	Y	N	**-**
May 1, 2020	June 5, 2020	N	Y	Marked 6/5/2020, 6/15/2020, 6/20/2020, 6/25/2020 Promise(s) as Cancelled by system
May 1, 2020	June 15, 2020	N	Y	
May 15, 2020	June 20, 2020	N	Y	
May 20, 2020	June 25, 2020	N	Y	

i.e., Irrespective of the promise taken date, if the system is going to mark the promise a broken on the current GL date processing, then system will mark the remaining active promises as per the new system parameter.

Example2:

GL date is June 1, 2020

Promise Amount: 100

No payment transactions on June 1, 2020

EOD, when the SET-COL CPPPRC_BJ_100_01 batch job gets executed

Parameter Value		BROKEN			
Taken Date	Promise Date	Collected	Broken Ind	Cancelled Ind	System action
April 12, 2020	May 6, 2020	100	N	N	-

April 24, 2020	May 12, 2020	100	N	N	-
May 1, 2020	June 1, 2020	0	Y	N	-
May 1, 2020	June 5, 2020	-	Y	N	Marked 6/5/2020, 6/15/2020, 6/20/2020, 6/25/2020 Promise(s) as broken by system
May 10, 2020	June 15, 2020	-	Y	N	
May 20, 2020	June 20, 2020	-	Y	N	
May 20, 2020	June 25, 2020	-	Y	N	

Note: Manually cancelled any specific promise will not automatically cancel/break future promises

Example 3:

GL date is June 1, 2020

Promise Amount: 100

No payment transactions on June 1, 2020

EOD, when the SET-COL CPPPRC_BJ_100_01 batch job gets executed, before that, June 20, 2020 promise was cancelled manually, then system cancels only remaining 3 promises. If the parameter is Broken, system does not mark/change the June 20, 2020 promise as broken

Parameter Value		CANCEL		
Taken Date	Promise Date	Broken Ind	Cancelled Ind	**System action**
May 1, 2020	June 1, 2020	Y	N	-
May 1, 2020	June 5, 2020	N	Y	Marked 6/5/2020, 6/20/2020, 6/25/2020 Promise(s) as Cancelled by system
May 1, 2020	June 15, 2020	N	Y	
May 15, 2020	June 20, 2020	N	Y (manual)	
May 20, 2020	June 25, 2020	N	Y	

Example 4:

GL date is June 1, 2020

Promise Amount: 100

On May 28th user cancelled the June 1, 2020 promise manually

EOD, when the SET-COL CPPPRC_BJ_100_01 batch job gets executed, then system does not take any action on remaining promises. If the parameter is Broken, system does not mark/change broken indicator as Y for remaining promises i.e., 6/5/2020, 6/15/2020, 6/20/2020, 6/25/2020 promises remain active

Parameter Value	Broken

Taken Date	Promise Date	Broken Ind	Cancelled Ind
May 1, 2020	June 1, 2020	N	Y (manual)
May 1, 2020	June 5, 2020	N	N
May 1, 2020	June 15, 2020	N	N
May 15, 2020	June 20, 2020	N	N
May 20, 2020	June 25, 2020	N	N

GL date is **June 5, 2020**

Promise Amount: 100

No payment transactions on June 5, 2020 or payment was not satisfied

EOD, when the SET-COL CPPPRC_BJ_100_01 batch job gets executed

Parameter Value		Cancel	
Taken Date	Promise Date	Broken Ind	Cancelled Ind
May 1, 2020	June 1, 2020	N	Y (manual)
May 1, 2020	June 5, 2020	Y	N
May 1, 2020	June 15, 2020	N	Y
May 15, 2020	June 20, 2020	N	Y
May 20, 2020	June 25, 2020	N	Y

Impact of change:

System will automatically update broken promise record counter and same will reflect on screen at the below path:

(Customer service → Summary → Delinquency information → Broken Promises (Life) / Broken Promises (Year))

I.e. if first promise is broken, and there are future 4 more promises, the counter will the value 5.

Even if the customer makes partial payment from the first Promised Amt and promise indicator is recorded as Broken, system would automatically record the future promises as broken. i.e., existing checks and validations of treating the current date promise is broken will continue.

2.4.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.5 Export to Excel

2.5.1 Overview

- ‘Export to excel’ facility is provided in selected screens to enable the export of data to spreadsheet.
- Also, screens enhanced to display the count of records in corresponding screens.

2.5.2 Description

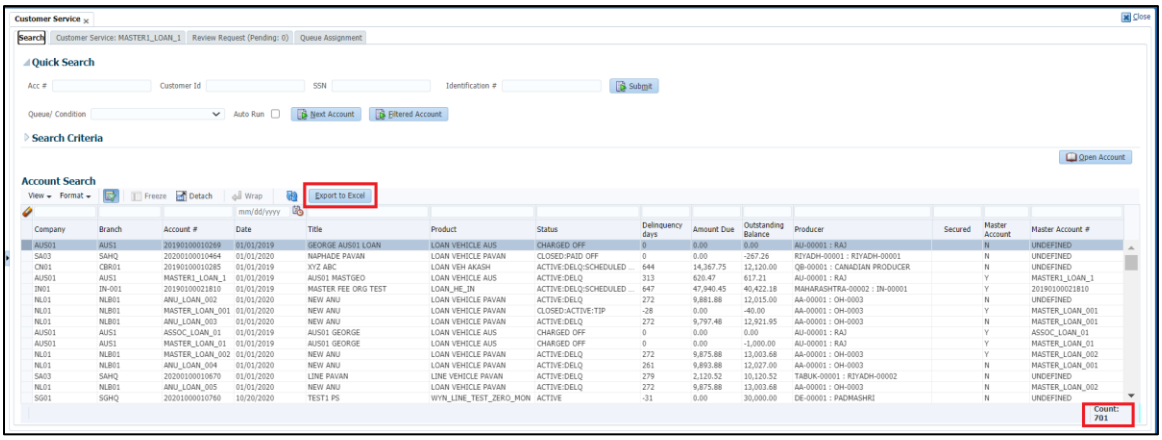
Export to Excel: Provided ‘Export to Excel’ button and Count details in following screens. ‘Export to Excel’ button provided with access key controls.

Customer Service

- Queues/Search Results- origination, Servicing, Collections
- Accounts Header
- Balances
- Call Activities
- Maintenance
- Promises
- Due Date History
- Collateral
- Tracking Attributes
- Condition Details/ Condition/Queue History

Count of records:

- System displays the Count of Records in the Table. In case fetch size is 1000 to display on User Interface. It is not count of all the records in the Database
- **Example**: There can be 50,000 records in database if UI is fetching 1,000 records count will show only 1,000.



2.5.3 Seed Data

Refer ‘Appendix: Seed Data’ chapter.

2.6 New Lease Letters

2.6.1 Overview

Following new Letters are added to 'Lease Module' and would be triggered based on the specific event.

- Payoff Quote Letter [lcspoq_em_100_01]
- Paid-in Full Letter [lcspdf_em_111_01]

2.6.2 Description

Payoff Quote Letter:

This letter would be generated, if the user posts the transaction 'Payoff Quote' through Servicing→ Customer Maintenance Screen and selecting Payoff Quote Letter = Y in transaction parameters.

Paid-in Full Letter:

This letter would be generated, if the account is paid in full.

2.6.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.7 Payment Interface Enhancements

2.7.1 Overview

OFSLL existing payment Webservice and file upload has been enhanced to support following Actions:

- **Post**: Payment will be posted.
- **Reverse**: Payment will be reversed successfully through this action.
- **Void**: Posted Payment record will be voided.

2.7.2 Description

- A new parameter “Action” has been added in the header section of the following payment interfaces
  - Customer/Business Payment File Upload
  - Customer/Business Payment File Web Service
  - Customer Account File Upload
- New parameter Action will only support POST, VOID and REVERSE as action code. “Invalid Action Code” error message will be displayed if any other Action code is specified.
- Payment Posting/Reversal/Void processing for the above channels is enhanced to handle below validations

- Post:
  - Existing validation will be continued as it is.
  - Existing comment posting will continue as it is.
- Reverse/Void:
  - System validates for the Customer/Business number provided and display error message “Invalid Customer/Business Number”, if invalid customer/business is provided.
  - System validates for the Master Account number provided and display error message “Invalid Master Account Number”, if invalid Master account number is provided.
  - System validates that payment details exist for Customer/Business Number, Master Account Number, Payment Amount, Payment Date and Company combination, else display error message “No payment details found for the given details”.
  - If payments are applied to multiple accounts using payment hierarchy, system will validate for all the accounts there will be an active transaction available based on Customer/Business Number, Master Account Number, Payment Amount, Payment Date and Company combination. If such payment details are not available, then system will display error message “No payment details found for the given details”.
  - System validates for payment details and if multiple payment details exist for input provided, then system will display error message as “Multiple Records found, please contact Customer service Agent”.
  - System successfully Reverses the payment and display message as “Payment has been reversed”, if correct payment details are provided.
  - System successfully Voids the payment and display message as “Payment has been voided”, if correct payment details are provided.
  - System posts comment “Payment has been reversed for the Payment Id < ID >” when payment is reversed successfully.
- System posts comment “Payment has been voided for the Payment Id < ID >” when payment is voided successfully.
- System allows to Reverse payment successfully for an account in Status= Paid Off/Tip

- Existing payment file upload batch jobs IPUPRC_BJ_100_01 and IPCPRC_BJ_100_01 have been enhanced to support Void and Reversals.
- System posts comment “Payment has been reversed/voided for the Payment Id < ID >” when payment is reversed/voided successfully.

2.7.3 Seed Data

Refer ‘Appendix: Seed Data’ chapter.

2.8 Event Enhancement (Added Securitization Event Type)

2.8.1 Overview

OFSLL has been enhanced to support Events framework for new entity securitization.

List of transaction codes supported are as below:

SPL_ADD_ACC [ADD ACCOUNT]

SPL_REMOVE_ACC [REMOVE ACCOUNT]

2.8.2 Description

- A new entity “SECURITIZATION” has been added in events framework.
- Following new events types have been added to the lookups.
  - SEC_CREATE
  - SEC_UPDATE
- Following new user defined tables have been added.
  - EVENT_SEC_PARAMETER_CD
  - EVE_SEC_CREATE_RS_PARAMETER_CD
  - EVE_SEC_UPDATE_RS_PARAMETER_CD
- Following attributes has been added under user defined tables created for new entity securitization.
  - ACC_PRODUCT_TYPE_CD
  - ACC_FUNDING_TYPE_CD
  - ACC_NBR
  - ACC_POO_POOL
- Events created for securitization has been supported for both ONLINE and BATCH.
- This event support only Webhook Action.

2.8.3 Seed Data

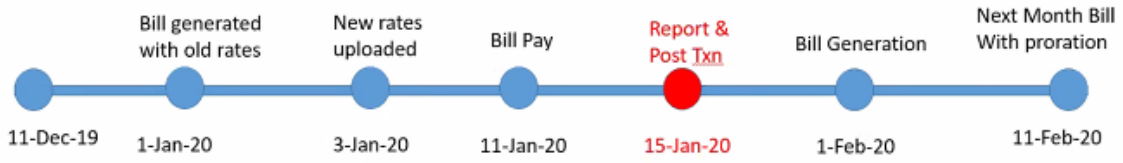
Refer ‘Appendix: Seed Data’ chapter.

2.9 Reporting Function – Due Amount between Due Dates

2.9.1 Overview

This enhancement supports function for calculating Due Amount for the specific Due Date and Account Number provided. This function supporta returning Due amount for past and future dues.

The function is required to identify the due amount that need to be posted on an account before the next dues is generated. For e.g.



Billing cycle maintained for an account A1 is monthly and is from 11-Dec-19 to 11- Jan-20. Based on pre-bill days, the bill is generated on 1-Jan-20 for dues till 11-Jan-20.

Now the Account A1 is Traded and new account is created on 3-Jan-20. Or Now the New rates are uploaded on 3-Jan-20.

The function will provide future due D1 for Due Date 11-Feb-20 and Past due D2 for 11-Jan-20.

Hence Proration amount = D1- D2 i.e. the difference between the future and past dues which need to be posted on Account before the next due is generated.

2.9.2 Description

- This enhancement support functions for providing Due amount as per the input provided. These functions are accessible from BI Publisher reports.
- The Function will accept Account Number and Date as input parameters and will provide Due Amount for provided due date as output.

Business Flow:

The Reporting function is used as per below process:

- The identified Accounts and the Future Due date for which due is required need to be provided as Input file to OFSLL.
- A new batch job IADPRC_BJ_100_01 under SET->IFP has been added to process the Input File and calculate the Dues for Account Number and Due Date provided.
- Reporting function is used to provide Future and past dues.

Data Files:

A new Input File Definition has been provided in “Account Due Date Upload” in Setup->Data Files->Input Definition. The column definition are as indicated below:

- Account Number
- Due Date

Batch Job:

- A new batch job has been created under SET->IFP.
- The batch job will calculate Due Amount for the Account Number as per the Due Date provided in Input File Definition “Account Due Date Upload” and store the computed values in new tables.
  - The Input file will be processed as per System parameter CMN_FILE_PROCESS_TO_LOB
  - The batch job would return regular bill amount calculated without considering any other outstanding dues. For e.g. Delq Due Amount, Late Fee etc.
  - If Account Number or Due Date is not provided, then display error message “Mandatory field <<< Field Name >>> missing”.
  - If the Account Number provided does not exist, then display error message “Invalid Account Number”.
  - If the Account Number provided status <> Active, then display error message “Invalid Account Status”.
  - If Due Date provided does not match with Due date of account, then display error message “Invalid Due Date provided”.
  - If the Due is already generated on account for the Due Date provided, then display error message “Due Amount already exist”.
  - System will generate and write above validations, if received during record processing in a Bad File.
  - If a valid Future Due date and Account Number is provided and record already exist in the Table, then batch job will calculate and replace the existing entries.

Function:

The Function-(**cmnval_cl_000_02.get_amount_due ()**) will return output on basis of below logic:

- The Function will accept following input parameter and provide Due Amount as result:
  - Input Parameter:
    - Account Number
    - Date
  - Output
    - Due amount for Due Date provided
- If the due is generated for Due date provided, then system will verify:
  - If Due Date provided is in last 4 due dates, then provide value accordingly from ACC_DUE_AMT1, ACC_DUE_AMT2, ACC_DUE_AMT3 and ACC_DUE_AMT4.
  - If Due Date provided is greater than last 4 due dates, provided, then the function will refer Due Date History for Due Amount where ADH_PRIMARY_IND =Y.
- If the Due Date provided is future dated and Account Status in Active, then the function will provide Due amount for Acc # as on Due date provided while referring to above tables that are updated by new batch job for calculating Due amount as per “Account Due Date Upload” Input File Definition setup.
  - If the new table does not have data for the date and account provided, then system will return function value as 0
- If Account Number provided in input does not exist, then function will return 0.

- If Input Date provided does not match with Accounts past or future Due Dates, then function will return value as 0.

2.9.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.10 Rescission Transaction Enhancement

2.10.1 Overview

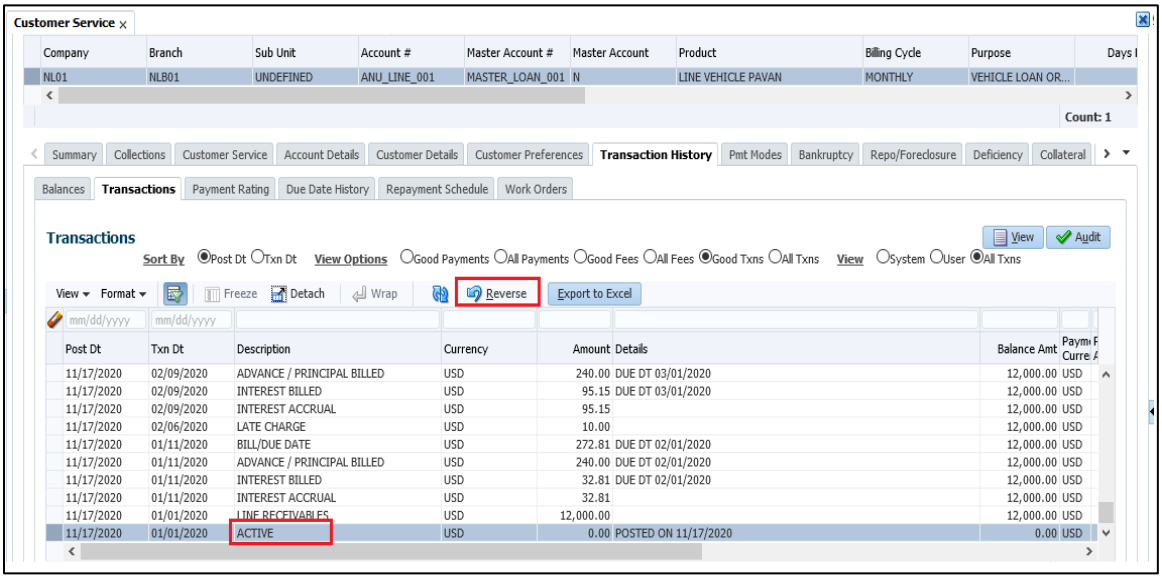
Rescission/Void transaction has been enhanced to support posting RECISSION on converted accounts. RESCISSION transaction processing on converted account will be same as standard account VOID functionality

- All the account balance will become 0 and total amount will get reversed, that amount will be shown in the rescission transaction.
- All the transaction amount will become 0.
- Account status will become VOID.

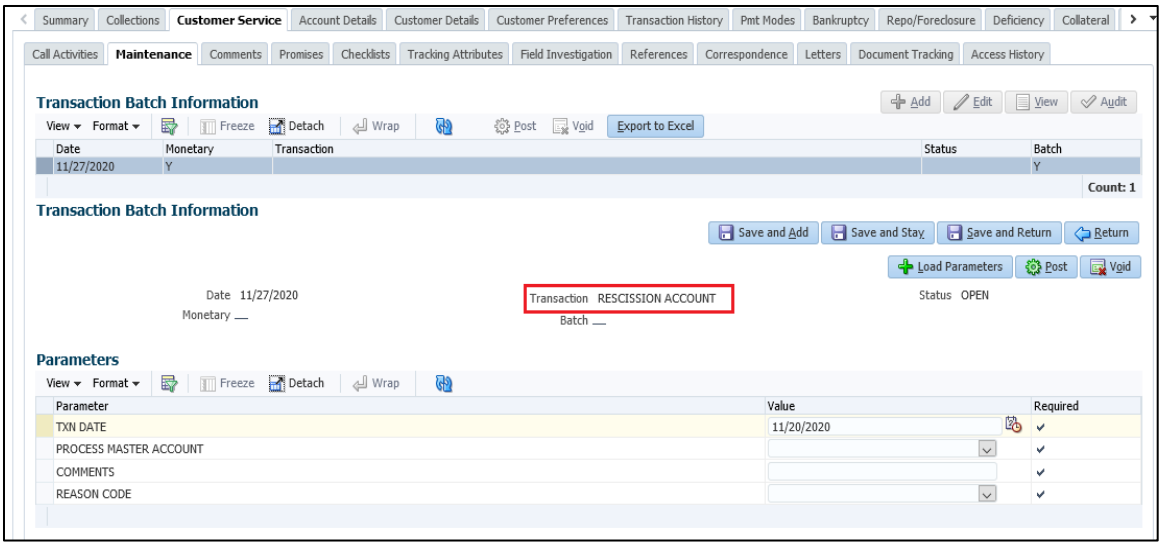
2.10.2 Description

- A new system parameter AUTO_GEN_ACTIVE_TXN_CONV has been introduced as part of the migration process to indicate if system needs to create a dummy ACTIVE transaction only for conversion accounts.
  - If the parameter is set to YES, migration batch job has been enhanced to insert a new dummy ACTIVE transaction record with the transaction date as conversion date.
  - During insertion, both TXN_PRIMARY_IND and TXN_BACKDATE_ALLOWED_IND has assigned constant value as YES to enable RESCISSION /VOID posting for the migrated account.
- System is enhanced to allow RESCISSION /VOID posting for migrated accounts provided ACTIVE transaction with TXN_PRIMARY_IND = Y is available in OFSLL for a migrated account.
- User will be allowed to post the Rescission transaction for a migrated account from customer maintenance screen or from customer service > transaction history by selecting the dummy ACTIVE transaction and reversing the same.

Rescission Transaction from Transaction History screen

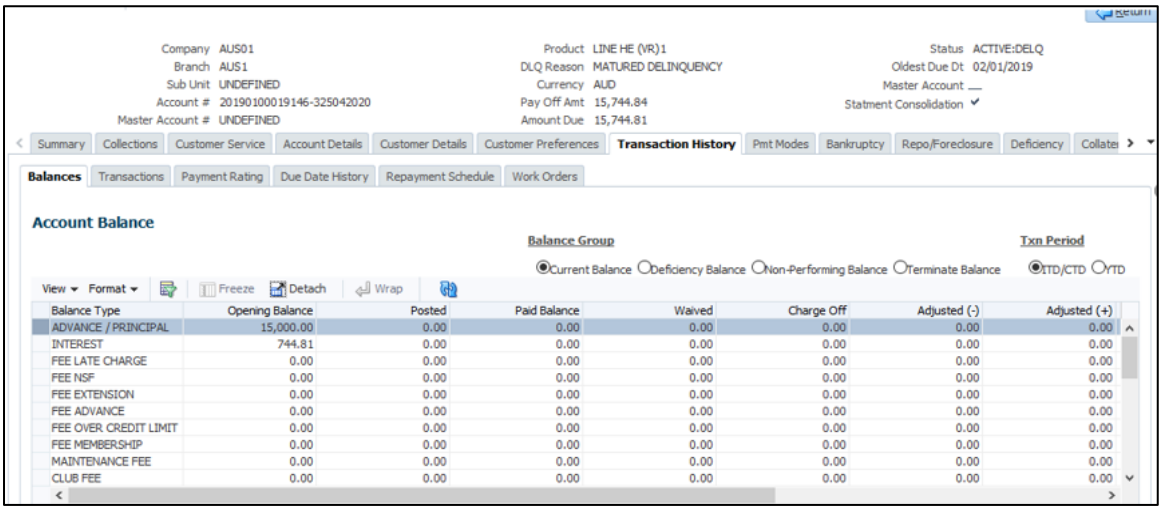


Rescission Transaction from Customer Maintenance screen (Customer Service >Customer Service > Maintenance)

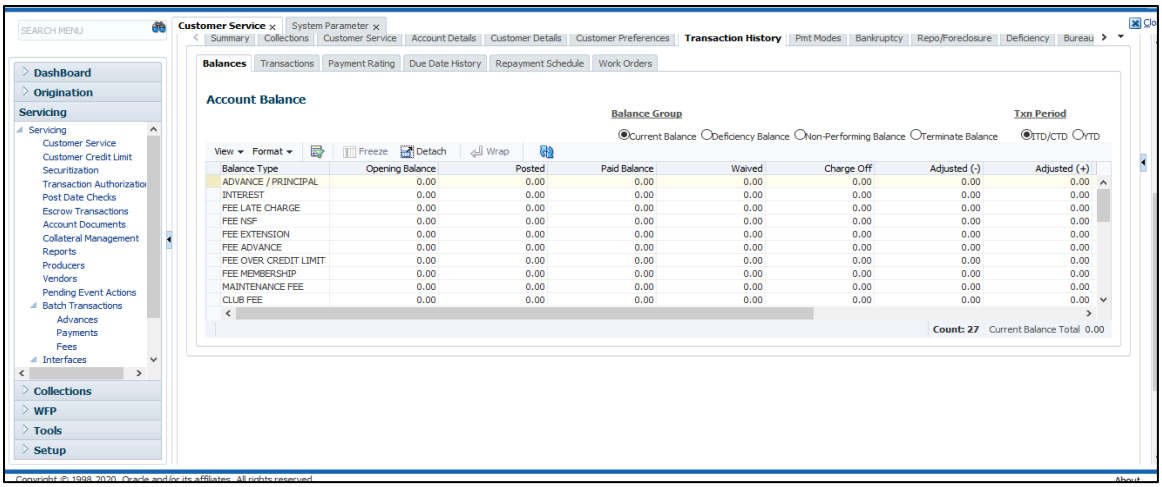


- VOID/ RESCISSION posting on migrated account will update all balances to zero. Any Good Transactions that were posted in OFSLL after the ACTIVE transaction will be reversed part of VOID/ RESCISSION process.
- During VOID/RESCISSION for a conversion account, if there is no transaction record other than the dummy ACTIVE transaction, system will post REVERSE ACTIVE transaction and update all balances to zero.

Before Rescission



After Rescission – Once Reverse ACTIVE transaction is posted, the balances will be updated to zero. No GL entries corresponding to legacy transaction balances will be posted in OFSLL.



- As per the current functionality, OFSLL will pass GL reversals for the transactions that are posted with TXN_PRIMARY_IND as Y in OFSLL. Legacy accounts does not have this indicator as Y to avoid GL entries.
- The above changes will not impact legacy accounts which were migrated without ACTIVE transaction as new process will not allow posting VOID/ RESCISSION on any migrated account that does not have ACTIVE transaction in system.

2.10.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.11 Statements Enhancements

2.11.1 Overview

Below mentioned existing statements has been modified to include account currency and product code.

SI NO	Statement Name	Description
1	CUSTOMER_BUSINESS_STATEMENT	CUSTOMER/BUSINESS STATEMENT
2	MASTER_CUSTOMER_BUSINESS_STMT	MASTER CUSTOMER /BUSINESS STATEMENT
3	MASTER_ACCOUNT_MOCK_STATEMENT	MASTER ACCOUNT CUSTOMER MOCK STATEMENT

2.11.2 Description

2.11.2.1 Customer/Business Statement

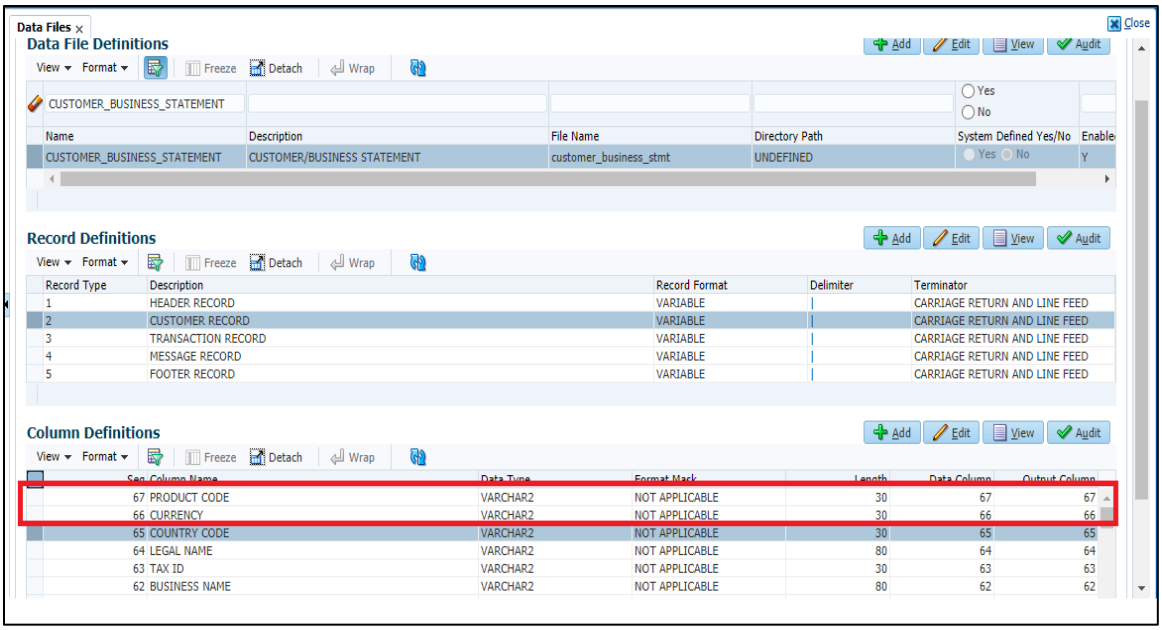
Data File Definition:

Below mentioned data file definition has been modified to include account currency and product code.

SI NO	Data File Definition Name	Description
1	CUSTOMER_BUSINESS_STATEMENT	CUSTOMER/BUSINESS STATEMENT

Below additional columns has been added to record type Customer Record (Record Type 2) to capture the account currency and product code.

SEQ NO	Record Type	COLUMN NAME	COLUMN/DATA/SORT SEQ	DATA TYPE	LENGTH
1	2	CURRENCY	66	VARCHAR2	30
2	2	PRODUCT CODE	67	VARCHAR2	30



Batch Job:

- Existing Customer Statements Generation ('OSTPRC_BJ_100_01') Batch jobs business logic has been enhanced to fetch account currency and product code.
- Fetched currency and product code value has been populated on to corresponding columns as defined in the data file definition above.

2.11.2.2 Master Customer/Business Statement

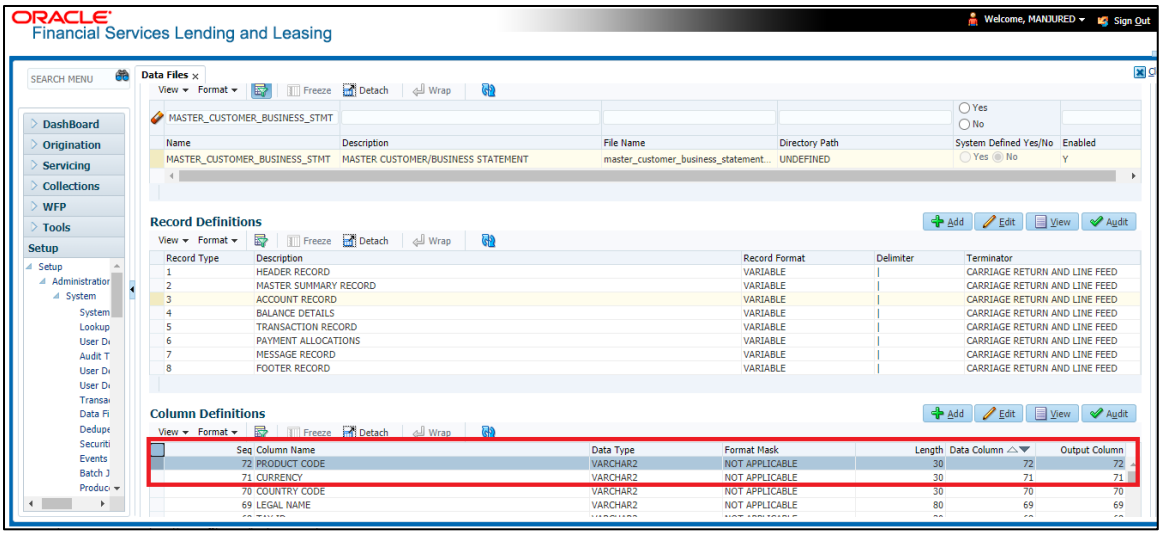
Data File Definition:

Below mentioned data file definition has been modified to include account currency and product code.

SI NO	Data File Definition Name	Description
1	MASTER_CUSTOMER_BUSINESS_STMT	MASTER CUSTOMER/ BUSINESS STATEMENT

Below additional columns has been added to record type 3 (Account Record) to capture the account currency and product code.

SEQ NO	Record Type	COLUMN NAME	COLUMN/DATA /SORT SEQ	DATA TYPE	LENGTH
1	3	CURRENCY	71	VARCHAR2	30
2	3	PRODUCT CODE	72	VARCHAR2	30



Batch Job:

- Existing Master Customer Statement Generation (‘OSTPRC_BJ_100_02’) Batch jobs business logic has been enhanced to fetch account currency and product code.
- Fetched currency and product code value has been populated on to corresponding columns as defined in the data file definition above.

2.11.2.3 Master Account Mock Statement

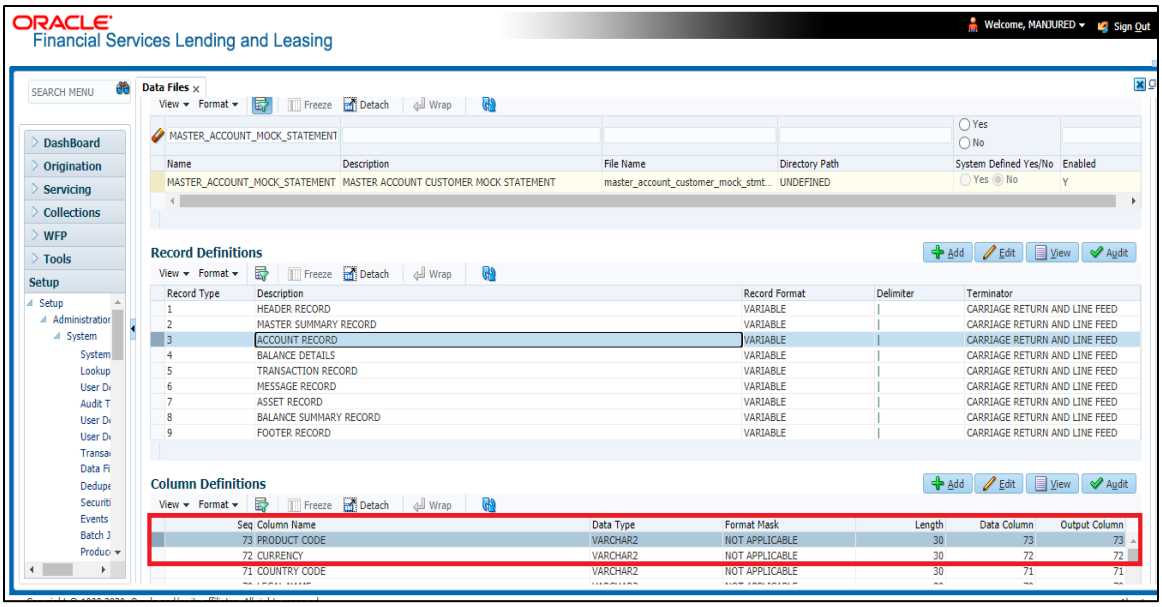
Data File Definition:

Below mentioned data file definition has been modified to include account currency and product code.

SI NO	Data File Definition Name	Description
1	MASTER_ACCOUNT_MOCK_STATEMENT	MASTER ACCOUNT CUSTOMER MOCK STATEMENT

Below additional columns has been added to record type 3 (Account Record) to capture the account currency and product code.

SEQ NO	Record Type	COLUMN NAME	COLUMN/DATA/ SORT SEQ	DATA TYPE	LENGTH
1	3	CURRENCY	72	VARCHAR2	30
2	3	PRODUCT CODE	73	VARCHAR2	30



Batch Job:

- Existing Master Account Customer Mock Statement Generation (‘OMSPRC_BJ_100_01’) Batch jobs business logic has been enhanced to fetch account currency and product code.
- Fetched currency and product code value has been populated on to corresponding columns as defined in the data file definition above.

2.11.3 Seed Data

Refer ‘Appendix: Seed Data’ chapter.

2.12 Mock Statement Enhancements

2.12.1 Overview

Existing 'MASTER ACCOUNT CUSTOMER MOCK STATEMENT' Data File definition has been modified to include 'BALANCE SUMMARY RECORD' record definition.

Existing 'OMSPRC_BJ_100_01' (MASTER ACCOUNT CUSTOMER MOCK STATEMENT GENERATION) batch job business logic has been enhanced to build and store the Balance Summary Record as per the record definition. System would sum up common balances from Master and Associated account and make it available in this Balance Summary record.

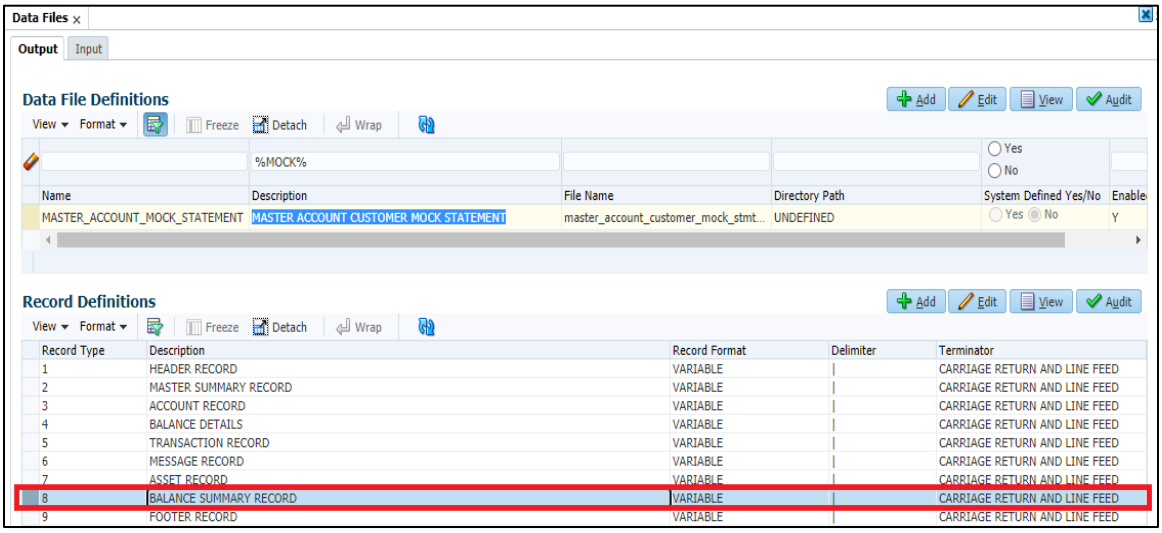
2.12.2 Description

Master Account Mock Statement Data File Definition:

Below mentioned data file definition has been modified to include Balance Summary record.

SI NO	Data File Definition Name	Description
1	MASTER_ACCOUNT_MOCK_STATEMENT	MASTER ACCOUNT CUSTOMER MOCK STATEMENT

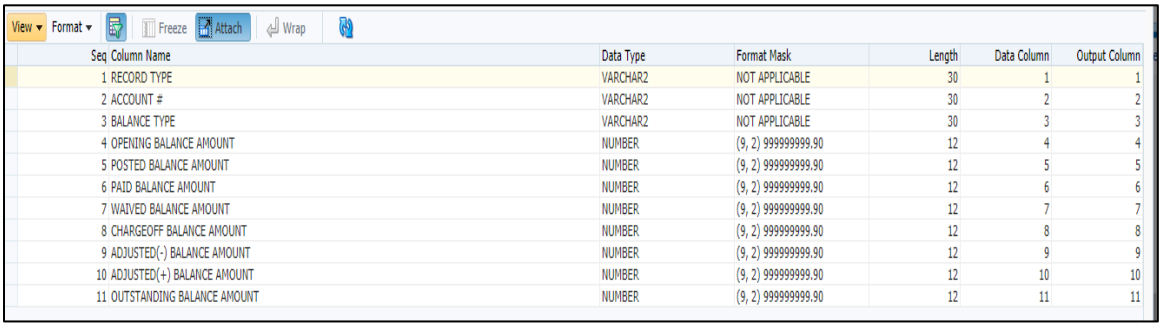
Balance Summary Record has been added with Record Type '8' as shown below.



All the below column definitions has been added as part of the Balance Summary Record.

SEQ NO	COLUMN NAME	DATA TYPE	LENGTH
1	RECORD TYPE	VARCHAR2	30
2	ACCOUNT #	VARCHAR2	30
3	BALANCE TYPE	VARCHAR2	30
4	OPENING BALANCE AMOUNT	NUMBER	12

5	POSTED BALANCE AMOUNT	NUMBER	12
6	PAID BALANCE AMOUNT	NUMBER	12
7	WAIVED BALANCE AMOUNT	NUMBER	12
8	CHARGEOFF BALANCE AMOUNT	NUMBER	12
9	ADJUSTED(-) BALANCE AMOUNT	NUMBER	12
10	ADJUSTED(+) BALANCE AMOUNT	NUMBER	12
11	OUTSTANDING BALANCE AMOUNT	NUMBER	12



Seq	Column Name	Data Type	Format Mask	Length	Data Column	Output Column
1	RECORD TYPE	VARCHAR2	NOT APPLICABLE	30	1	1
2	ACCOUNT #	VARCHAR2	NOT APPLICABLE	30	2	2
3	BALANCE TYPE	VARCHAR2	NOT APPLICABLE	30	3	3
4	OPENING BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	4	4
5	POSTED BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	5	5
6	PAID BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	6	6
7	WAIVED BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	7	7
8	CHARGEOFF BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	8	8
9	ADJUSTED(-) BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	9	9
10	ADJUSTED(+) BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	10	10
11	OUTSTANDING BALANCE AMOUNT	NUMBER	(9, 2) 999999999.90	12	11	11

As you can notice from the Fig 2.11.2.1 the Record type of Footer record has been modified from 8 to 9 to include Balance summary record before the footer record.

Master Account Mock Statement Batch Job:

- Existing Master Account Statement Batch job ‘OMSPRC_BJ_100_01’ business logic has been enhanced to calculate and include Balance Summary record.
- System would summarize non-zero balance records for all accounts that satisfy below conditions
  - Master Account.
  - All it’s Associate Accounts that has Statement Consolidate Indicator (‘ACC_STMT_CONSOLIDATE_IND_CUR’) flag value as ‘Y’.
- Balance summary record would be summarized as shown below:

  Note: We have assumed that PAID BALANCE AMOUNT, WAIVED BALANCE AMOUNT, CHARGEOFF BALANCE AMOUNT, ADJUSTED(-) BALANCE AMOUNT and ADJUSTED(+) BALANCE AMOUNT value as Zero in this example. We have ignored those columns in this example.

Master Account Balance Record

Account Number	Master Account Number	Balance Type	OPENING BALANCE AMOUNT	POSTED BALANCE AMOUNT	OUTSTANDING BALANCE AMOUNT
WYN_SFW_BILL_ MASTER_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH2	40	280	280

WYN_SFW_BILL_ MASTER_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH3	60	420	420
WYN_SFW_BILL_ MASTER_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH5	100	700	700
WYN_SFW_BILL_ MASTER_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH7	100	700	700

Associated Account Balance Record

Note: Assuming that below account 'WYN_SFW_BILL_DUES_ACC_6928' has Statement Consolidate Indicator ('ACC_STMT_CONSOLIDATE_IND_CUR') value as 'Y'.

Account Number	Master Account Number	Balance Type	OPENING BALANCE AMOUNT	POSTED BALANCE AMOUNT	OUTSTANDING BALANCE AMOUNT
WYN_SFW_BILL_ DUES_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH2	40	280	280
WYN_SFW_BILL_ DUES_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH3	60	420	420
WYN_SFW_BILL_ DUES_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH5	100	700	700
WYN_SFW_BILL_ DUES_ACC_6928	WYN_SFW _BILL_ MASTER_ ACC_6928	FOTH6	100	700	700

Balance Summary Record:

Account Number	Balance Type	OPENING BALANCE AMOUNT	POSTED BALANCE AMOUNT	OUTSTANDING BALANCE AMOUNT

WYN_SFW_BILL_ MASTER_ACC_6928	FOTH2	80	560	560
WYN_SFW_BILL_ MASTER_ACC_6928	FOTH3	120	840	840
WYN_SFW_BILL_ MASTER_ACC_6928	FOTH5	200	1400	1400
WYN_SFW_BILL_ MASTER_ACC_6928	FOTH7	100	700	700
WYN_SFW_BILL_ DUES_ACC_6928	FOTH6	100	700	700

2.12.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.13 Payment Refund Reason Code

2.13.1 Overview

Payment Refund Transaction has been enhanced to support below functionalities:

- User will be allowed to input the Reason code while posting Refund from Payment Maintenance screen
- Monetary 'Payment Refund' transaction includes the Reason code parameter.
- Reason code will also be available while posting Payment Refund transaction from the following functionalities
  - Generic Transaction Post web service
  - Transaction Upload File.

2.13.2 Description

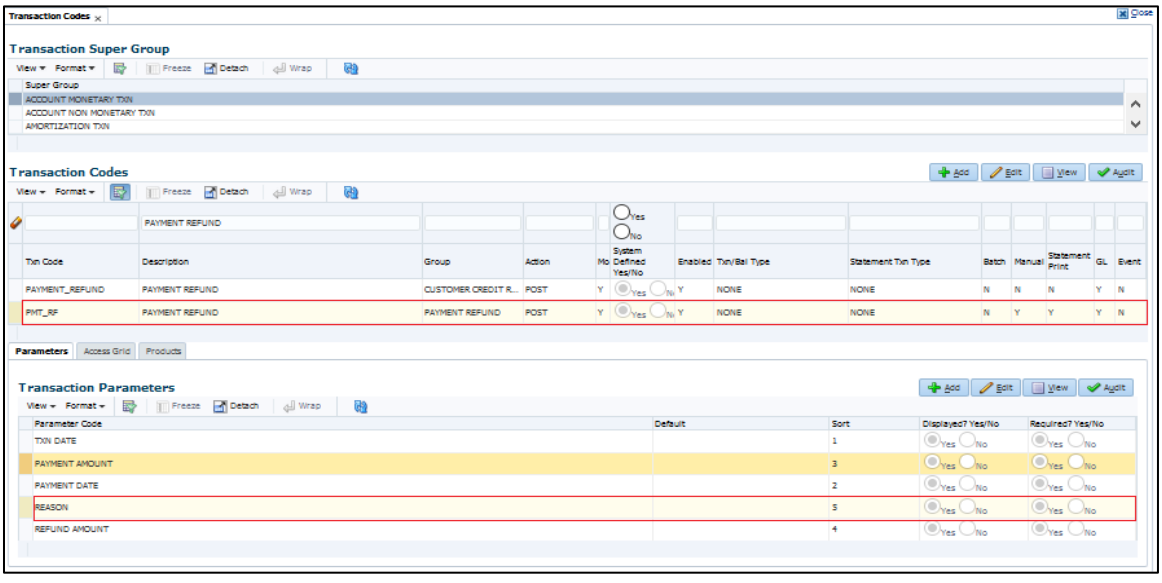
In order to include the Reason in the Payment Refund Transaction following changes has been done in OFSLL

Setup → Payment Refund → Transaction Parameter:

New Parameter 'Reason' has been added to Transaction Code: **PMT_RF (Payment Refund)** as shown below.

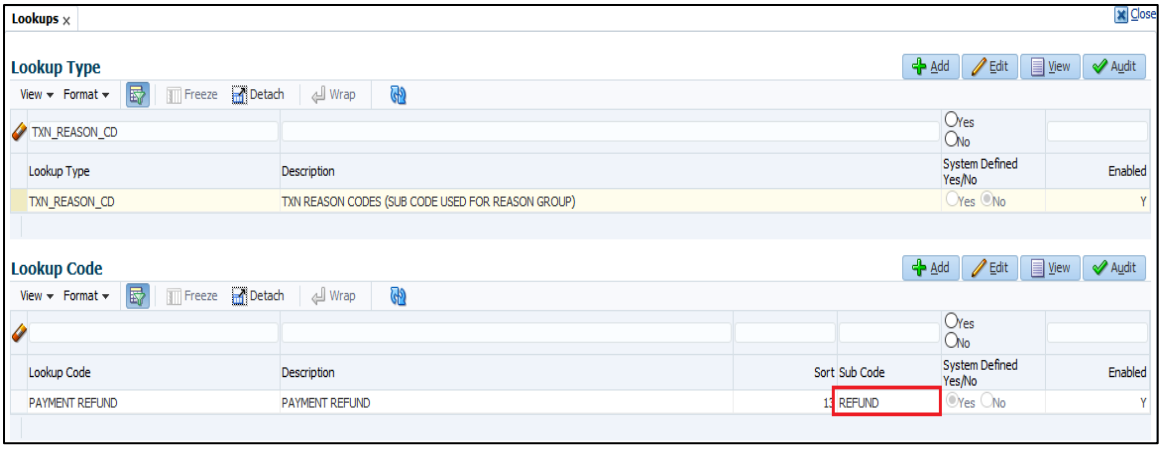
Parameter Code	Default	Sort	Displayed? Yes/No	Required? Yes/no
Reason		5	Yes	NO

Existing lookup TXN_REASON_CD has been used to select the appropriate reason codes from the Reason Drop down List.



Setup → System → Lookups:

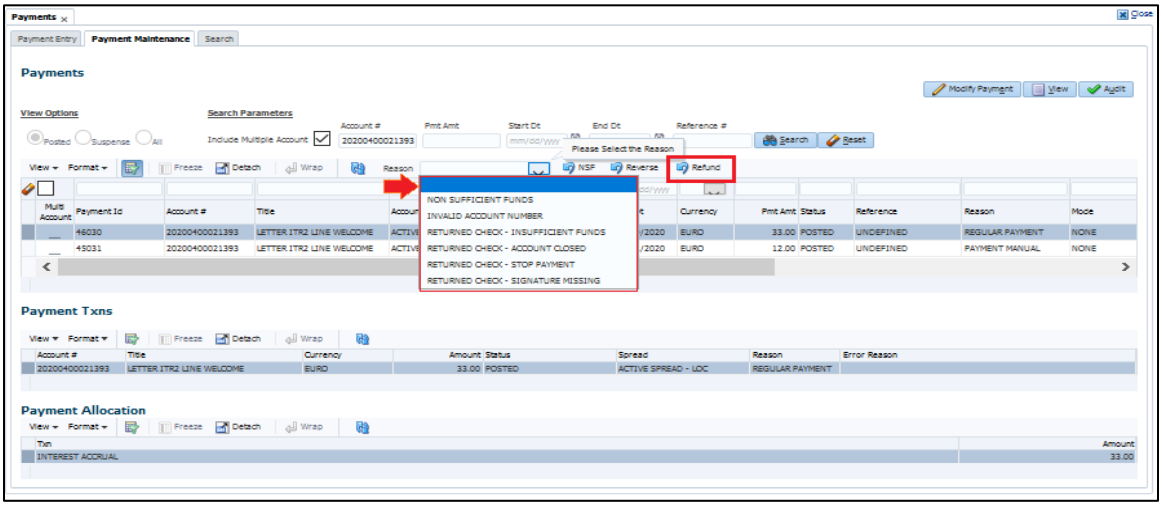
- New lookup sub-code 'REFUND' has been introduced under lookup type 'TXN_REASON_CD' to identify the reason codes that can be used for Payment Refund transaction.
- A new reason code 'PAYMENT REFUND' has been introduced with Lookup Type 'TXN_REASON_CD' and sub-code 'REFUND' which can be used for payment refund transaction.

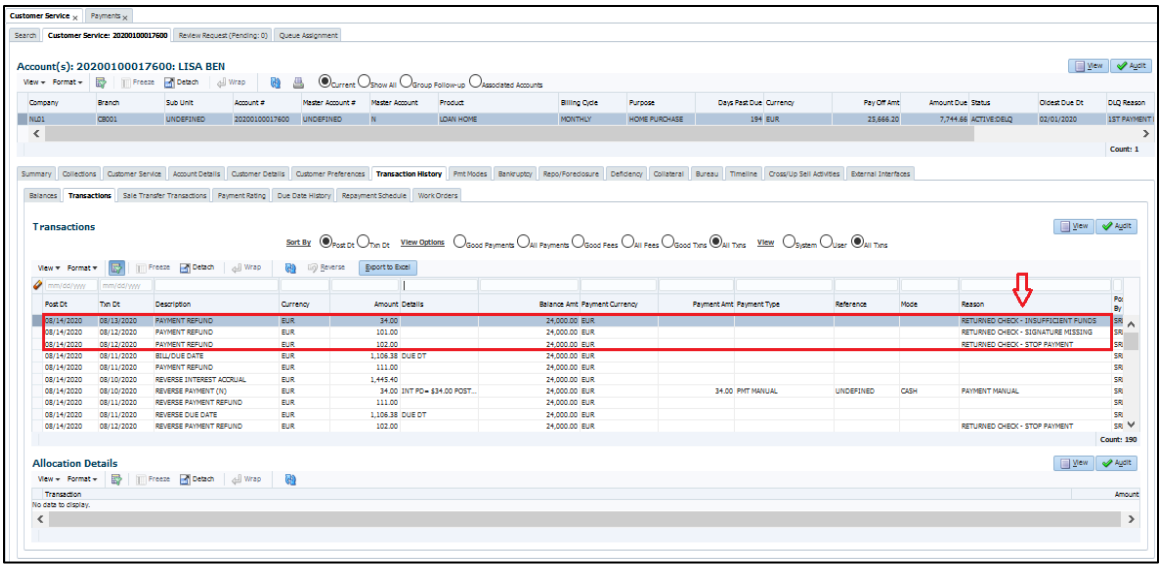


Customer Service → Payment Maintenance Screen:

OFSLL has been enhanced to support below functionality:

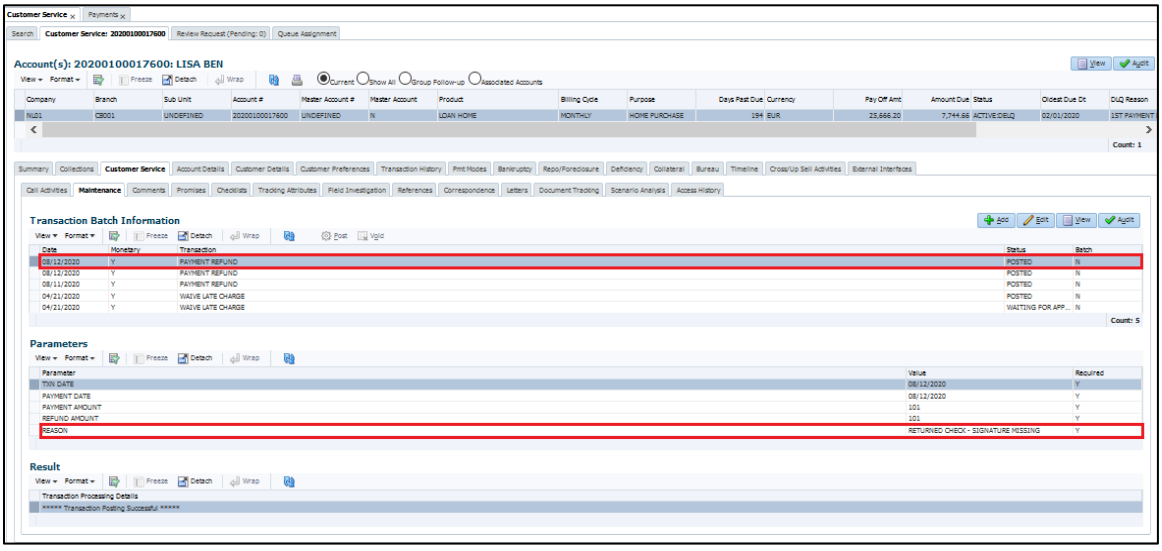
- On click on refund, system will successfully post the refund transaction.
- Reason code will get stamped on the transaction and will be visible under customer service ->transaction history ->transactions of the account.
- Existing query for Reason Code lookup in Payment Maintenance screen has been modified to include the reason codes for Payment Refund transaction as well (Reason Codes with sub-code as 'REFUND').
- A new validation has been introduced to ensure the reason code selected for Payment Refund transaction must belong to new sub-code 'REFUND', else error 'Invalid reason code selected for Refund' will be shown to the user.
- The above validation will be applicable for Payment Refund transaction posted from UI (Payment Maintenance, Customer Service>Maintenance), File Upload and Web Service.





Customer Service → Maintenance Screen:

- Similarly posting of 'Payment Refund Transaction' from Customer Service → Maintenance screen has been modified to include the 'Reason' parameter for user to capture the appropriate reason code.
- After posting, it would be visible under Customer Service →Transaction history →Transactions of the account as shown above.



Generic Transaction Post Web service:

- Reason code parameter is be available while invoking Payment Refund Transaction through Generic Posting Web service.
- Sample Payment Refund Transaction json file is attached.

Refund Payment Transaction.json

Transaction Upload File:

- Reason code parameter is be available while invoking Payment Refund Transaction file upload.
- Sample Refund Payment Transaction file is attached.

2.13.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.14 Asset Billing Enhancement

2.14.1 Overview

OFSLL existing Billing process has been enhanced so that the assets under Traded, Charged-Off and Voided accounts do not get considered for posting charges using Rate table.

2.14.2 Description

- A new lookup type **"ACC_STATUS_ASE_BILLING_CD**" (ACCOUNT STATUS FOR ASSET BILLING CODES) has been added to define the account status of the **Linked Account/Associated Account** that are to be considered for asset billing process part of the existing billing batch job (SET-TPE : TXNDDT_BJ_100_01) based on the **Due Amt Account Type.**
- Now, system includes the asset billing points of **Linked Account/Associated Account** only when the status of the **Linked Account /Associated Account** is in any one of the status maintained in the lookup type **"ACC_STATUS_ASE_BILLING_CD"**

2.14.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.15 Account UDF Update Service

2.15.1 Overview

A new Webservice has been created to support below functionalities:

- Update UDF details at account level.
- System will not validate if UDF has a formula configuration already available. I.e. updating of UDFs with formula will be allowed the same way it is currently allowed from UI.
- Description

New Web Services

Service Name	Purpose / Impact
Update Account UDF Service (Put)	New service to update udf of an existing account.

WebService Specifications:

- This WS will update the 50 UDF fields that are available in the account level.
- All the UDF input fields are non-mandatory but will require at-least one element to invoke this service
- Any UDF value(s) that are sent part of the request will be updated in the account level. UDF values not sent part of the request will remain unchanged.

Validations

The UDF input fields are non-mandatory, but at least one element for update is required to call this service. Else, system throws an error. User code and account number are validated. If invalid, system throws an error.

2.15.2 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.16 Same Producer Number across Company

2.16.1 Overview

OFSLL has been enhanced to use a producer across all branches of a given company, irrespective of producer's branch, during account boarding and from UI.

2.16.2 Description

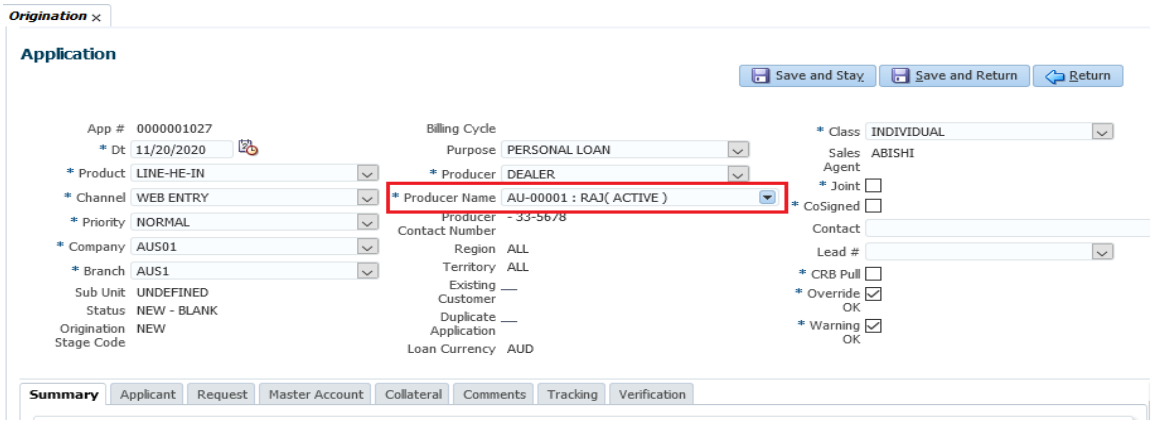
I. System has been enhanced to allow the use of any producer across all branches of the company to which the account belongs.

II. To implement the enhancement, during account on boarding the current branch validation which earlier prevented the producer to have any branch value other than the one specified to the account has been removed.

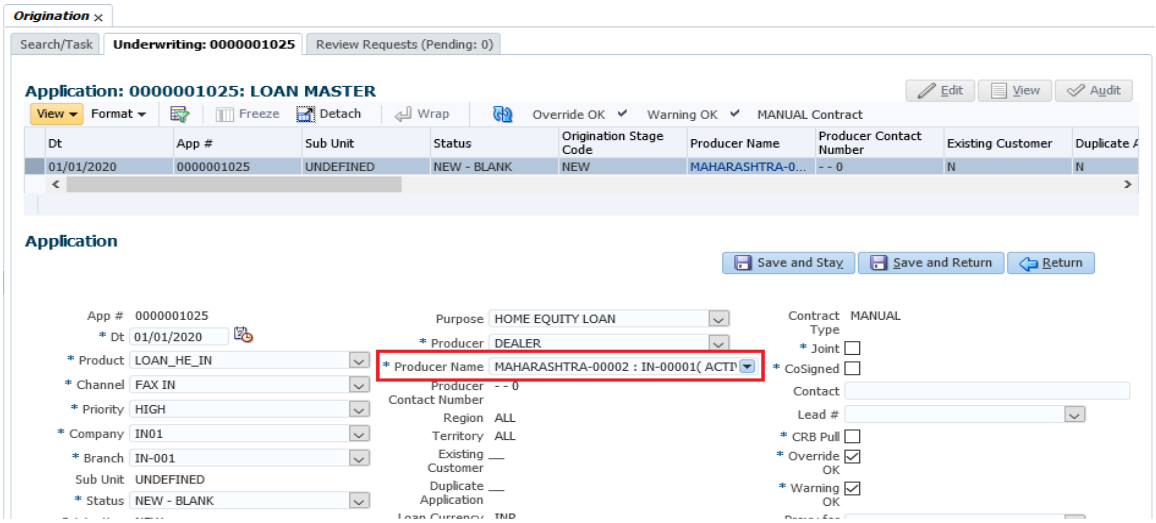
III. From the UI, the LOVs used for selecting a producer in the Simple Application entry, Application Entry, Underwriting, Funding and the Conversion Accounts screens have been enhanced to accept producer belonging to any branch of the same company.

IV. Additionally, an Origination Edit, Conversion Edit and an Account Boarding Edit have been introduced to check if the branch values of the producer and the account is same. In case of branch value mismatch, the edit will be raised. This edit is configurable and user can set this as a warning or error type.

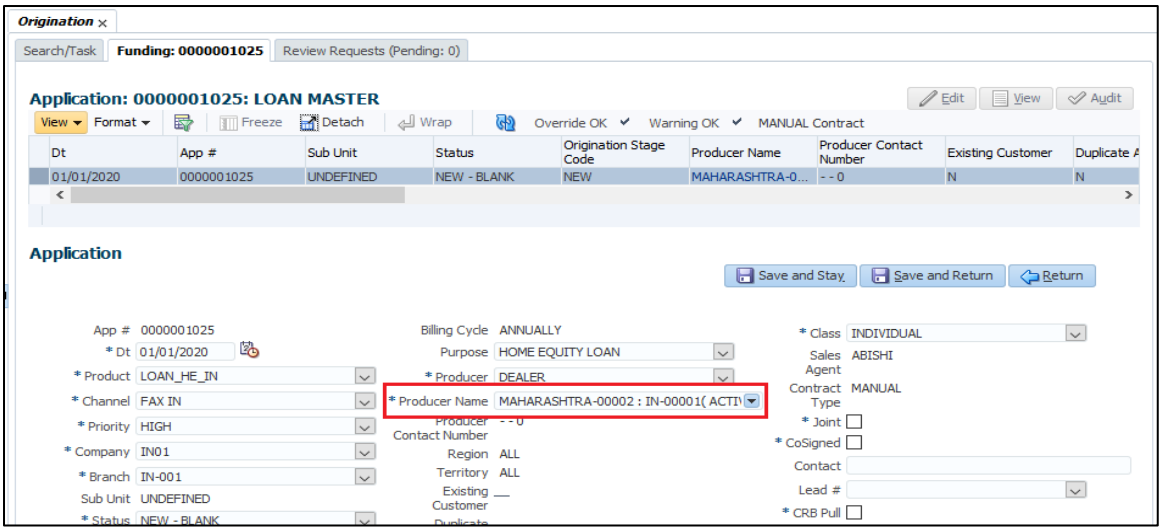
V. The following screenshot shows the LOVs that has been modified in Simple Application and Application Entry screens:



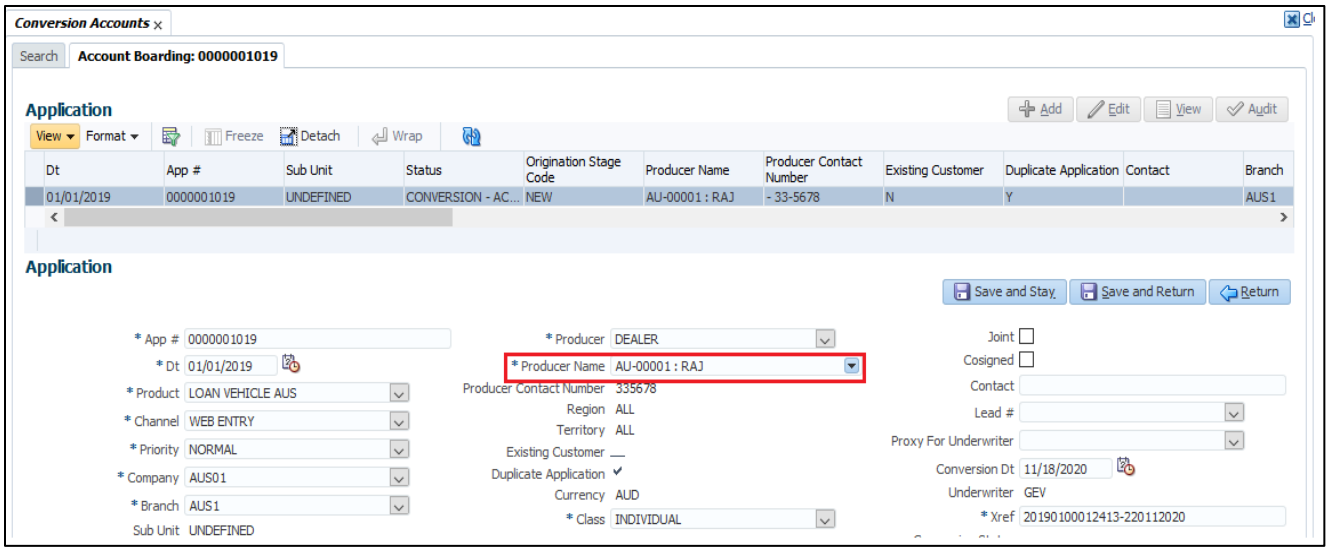
- The Underwriting screen:
  - Click on the edit button



- Funding screen:
  - Click on Edit



- Conversion Accounts:
  - Click on *Edit*



2.16.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.17 Charge off Transaction Enhancement

2.17.1 Overview

OFSLL has enhanced the existing '**CHARGED OF**F' Monetary Transaction as below:

- A provision has been made in the system to enable users to post CHARGED OFF transaction on the entire hierarchy of Associated/Linked Accounts along with the Master Account.

  i.e. If CHARGED OFF is initiated from an Associated/Linked account, system will be able to post CHARGED OFF on all ACTIVE associated/linked accounts in the hierarchy along with the Master Account.

- Previously system supported posting CHARGED OFF on Master Account only if Associated Accounts are in TRADED, VOID, TERMINATED status. CHGOFF and PAID statuses has been included in this list to allow processing CHARGED OFF on a Master Account in case any of the Associated/Linked Accounts are in CHARGED OFF/PAID status.

- System is enhanced to restrict posting Reversal of a CHARGED OFF transaction for an Associated Account if corresponding Master Account is not in ACTIVE status and shall display appropriate error message.

2.17.2 Description

CHARGED OFF Transaction Posting:

TXN Parameter	Description	Default Value	Display Indicator	Newly Added
Process Master Account	Posts CHGOFF transaction on Master Account	N	Y	N
Process Master Associated Account	Posts CHGOFF transaction on all Accounts under Master Account	N	Y	Y
Process Linked Account	Posts CHGOFF transaction on Linked Account	Y	N	N
Process Same Product Type and Funding Type Accounts	Posts CHGOFF transaction only on Accounts having same Product and Funding Type	N	Y	Y

- New Transaction Parameter **Process Master Associated Accounts** and **Process Same Product Type and Funding Type Accounts** has been introduced for CHARGED OFF Transaction. Below is the summary of parameters for CHARGED OFF transaction:

- While posting CHARGED OFF on Master Account, if the underlying the Associated Accounts are in below statuses, system allows posting CHARGED OFF on Master Account.

  PAID and CHGOFF statuses is included in the below list as part of this enhancement

  - TRADED
  - VOID
  - TERMINATE

- **PAID**
- **CHGOFF**

- If any error is encountered while processing CHARGED OFF for any of the accounts in the hierarchy, appropriate error is shown to the user and the entire transaction is rolled back.

CHARGED OFF transaction Reversal:

- A validation has been added as a part of this enhancement to ensure that CHARGED OFF reversal on Associated account is allowed only if Master Account is in ACTIVE status, otherwise, an error message will be shown to the user, stating –

'CHARGED OFF cannot be reversed - Master Account must be in ACTIVE status'.

Examples/Use cases:

- **Case -1**
  Master Accounts has two associated accounts as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	TRADED
ACC456	UNDEFINED	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the TRADED Account (ACC123)

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	TRADED
ACC456	UNDEFINED	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

- **Case – 2**
  Master Accounts has three associated accounts as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	VOID

ACC456	UNDEFINED	MAS123	ACTIVE
ACC789	UNDEFINED	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts =** N

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the VOID Account (ACC123).

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	VOID
ACC456	UNDEFINED	MAS123	CHGOFF
ACC789	UNDEFINED	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

- **Case – 3**

Master Account has three associated account as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	TERMINATE
ACC456	UNDEFINED	MAS123	ACTIVE
ACC101	UNDEFINED	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456.
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the TERMINATE Account (ACC123).

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	TERMINATE
ACC456	UNDEFINED	MAS123	CHGOFF
ACC101	UNDEFINED	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

- **Case -4**

Master Account has two associated account as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD

ACC123	UNDEFINED	MAS123	PAID
ACC456	UNDEFINED	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the PAID Account (ACC123).

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	PAID
ACC456	UNDEFINED	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

- **Case – 5**

Master Account has 4 associated accounts with accounts linked to each other as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	CHGOFF
ACC456	UNDEFINED	MAS123	ACTIVE
ACC789	ACC101	MAS123	ACTIVE
ACC101	ACC789	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N
- ACC123 is an associated account of MAS123 already in CHARGED OFF status.

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the Account (ACC123) which is already in CHARGED OFF status.

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	CHGOFF
ACC456	UNDEFINED	MAS123	CHGOFF
ACC789	ACC101	MAS123	CHGOFF
ACC101	ACC789	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

Scenario:

- CHARGED OFF Reversal posted on Associated account ACC123

Expected Behaviour- **System will throw an error since Master Account (MAS123) is still in CHARGED OFF.**

Scenario:

- CHARGED OFF Reversal posted on Master account MAS123

Expected Behaviour – The system will reverse the CHARGED OFF from the Master Account (MAS123).

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	CHGOFF
ACC456	UNDEFINED	MAS123	CHGOFF
ACC789	ACC101	MAS123	CHGOFF
ACC101	ACC789	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	ACTIVE

Now, CHARGED OFF Reversal can be posted on any of the accounts individually, since the Master Account is not in CHARGED OFF state.

Scenario:

- CHARGED OFF Reversal posted on Associated account ACC789

Expected Behaviour – The system will reverse the CHARGED OFF only from the Associated account ACC789.

Please note, even though ACC101 is linked with ACC789, the CHGOFF reversal on ACC789 will not impact account ACC101, which will remain in CHGOFF status and needs to be individually reversed.

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	UNDEFINED	MAS123	CHGOFF
ACC456	UNDEFINED	MAS123	CHGOFF
ACC789	ACC101	MAS123	ACTIVE
ACC101	ACC789	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	ACTIVE

- **Case – 7**

Master Account has 4 associated account as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	ACC456	MAS123	VOID
ACC456	ACC123	MAS123	ACTIVE
ACC789	ACC101	MAS123	ACTIVE
ACC101	ACC789	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456
- **Process Master Account** = Y
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behaviour - System will CHGOFF all ACTIVE accounts without impacting the VOID Account (ACC123).

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	ACC456	MAS123	VOID
ACC456	ACC123	MAS123	CHGOFF
ACC789	ACC101	MAS123	CHGOFF
ACC101	ACC789	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	CHGOFF

- **Case -8**

Master account has two associated accounts that are linked to each other.

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	ACC456	MAS123	ACTIVE
ACC456	ACC123	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: ACC456.
- **Process Master Account** = N
- **Process Master Associated Account** as = Y
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behaviour - System will CHGOFF all ACTIVE Associated accounts without impacting the Master Account.

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	ACC456	MAS123	CHGOFF
ACC456	ACC123	MAS123	CHGOFF
MAS123	UNDEFINED	MAS123	ACTIVE

- **Case-9**

  Master Account has 4 associated account as shown below:

ACC_NBR	ACC_LINKED_ACC_NBR	ACC_MASTER_ACC_NBR	ACC_STATUS_CD
ACC123	ACC456	MAS123	TRADED
ACC456	ACC123	MAS123	ACTIVE
ACC789	ACC101	MAS123	ACTIVE
ACC101	ACC789	MAS123	ACTIVE
MAS123	UNDEFINED	MAS123	ACTIVE

Scenario:

- CHARGED OFF posted on account: MAS123
- **Process Master Account** = Y/N
- **Process Master Associated Account** as = Y/N
- **Process Same Product Type and Funding Type Accounts** = N

Expected Behavior: System will throw an error since the associated accounts are in ACTIVE state.

As per the existing functionality when we post the Charged-off transaction from the Master Account, the **Process Master Account** and **Process Master Associated Account** has no role while carrying out the Charge off transaction. System just checks if all other Associated Accounts are in appropriate state, only then system will post CHARGED-OFF on the Master Account.

2.17.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.18 New Fee Calculation Method (Master Fee)

2.18.1 Overview

a. OFSLL has been enhanced to support two new Late Fee types “Cycle Based Collection Late Fee” and “Cycle Based Late Fee” for Loan/Line/Lease Account. These Fees have their own Billing Cycles which can be configured differently from Account Billing Cycle to enable charging Late Fee multiple times within a single Billing Period for Account.

b. Also, OFSLL has been enhanced to have the ability to calculate and post Consolidated Fee at Master Account for all/selected associated accounts for below Fee types:
   i. Late Charge
   ii. Cycle Based Collection Late Fee
   iii. Cycle Based Late Fee

2.18.2 Description

- **Setup Changes**

I. Following new fields have been added under new header “Fee Consolidation” [after the “Delinquency” header] in Setup >Contract to denote if Fee Consolidation at Master Account is enabled for the below Fees for Account.

   a. Late Charge at Master Account
   b. Cycle based Collection Late Fee at Master Account
   c. Cycle Based Late Fee at Master Account

Field Name	Mandatory Y/N	Data Type/ Size	Default Values
Late Charge at Master Account	Y	Check Box	N
Cycle based Collection Late Fee at Master Account	Y	Check Box	N
Cycle Based Late Fee at Master Account	Y	Check Box	N

II. Following fields have been added to capture Grace Days for new Cycle Based Late Fee Grace Days” under “Delinquency” header in Setup >Contract after Delq Grace Days field.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values
Cycle Based Late Fee	Y	Check Box	N
Cycle Based Collection Late Fee	Y	Check Box	N

Cycle based Collection Late Fee Grace Days	Y	Text Box (Number)	0
Cycle based Late Fee Grace Days	Y	Text Box (Number)	0

Below validations have been introduced in the system for the newly added fields

a. If “Cycle based Collection Late Fee at Master” is selected, then “Cycle based Collection Late Fee” flag also must be selected, else system will show an error message saying “Cycle based Collection Late Fee is to be selected” on save of the record.

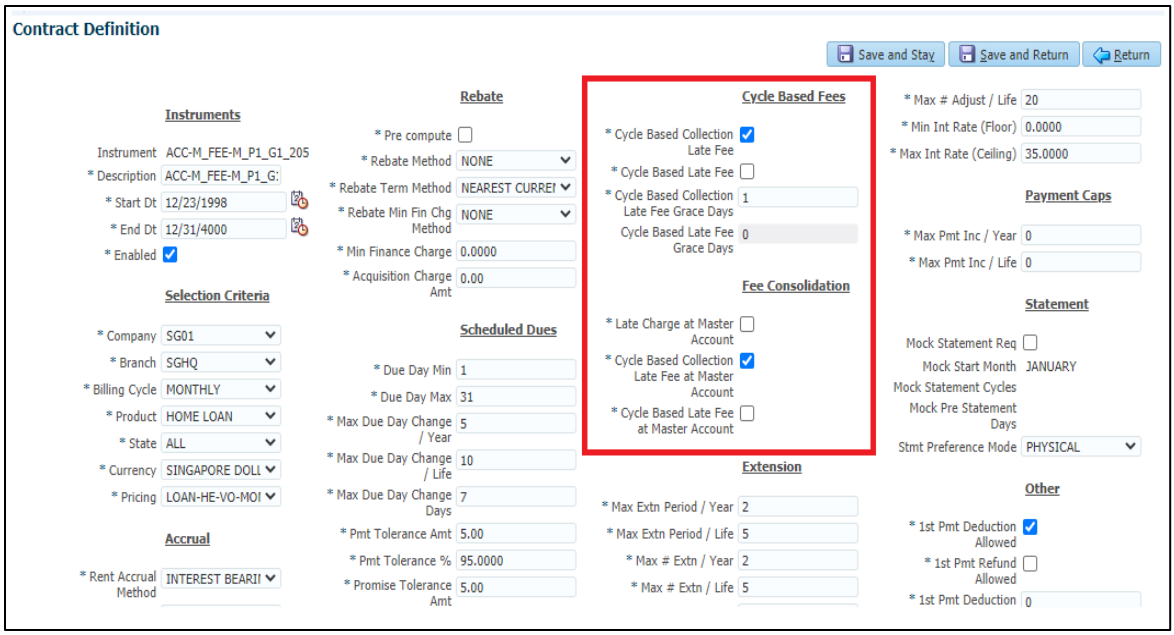
b. Similarly, if “Cycle Based Late Fee at Master” is selected, then “Cycle Based Late Fee” flag need to be also selected, else system will throw appropriate error message on save of the record.

c. If the Cycle based Collection Late Fee flag is not selected for an Account, then
   i. System will not load the “Cycle based Collection Late Fee” balance when user clicks on “Load Balances” button in Contract >Balances, even if the Product Type configured for respective transactions. (Similar to ACH Fee Ind functionality).
   ii. “Cycle based Collection Late Fee” will not be shown in the Lookup under Contract >Fees > Type.

d. If the Cycle based Collection Late Fee flag is selected for an Account, then
   i. “Cycle based Collection Late Fee” must be configured in “Contract >Fees”. If user tries to enable the Contract then on save, system will show an error message saying, “Cycle based Collection Late Fee not specified for the contract”.

e. Similarly, if the Cycle Based Late Fee flag is not selected for Account
   i. System will not load the “Cycle Based Late Fee” balance when user clicks on “Load Balances” button in Contract >Balances.
   ii. “Cycle Based Late Fee” will not be shown in Contract >Fees > Type lookup.

f. If the Cycle Based Late Fee flag is selected, then
   i. System will not allow to save the enabled Contract record without “Cycle Based Late Fee” configured in “Contract >Fees”. System will show an error message saying, “Cycle based Late Fee not specified for the contract”.



III. Following new Fee Calculation Methods have been added in “FEE_CALC_METHOD_CD” lookup for “FEE LATE CHARGE”.

a. PERCENTAGE OF SUM OF PAYMENT DUE
b. PERCENTAGE OF SUM OF STANDARD PAYMENT
c. PERCENTAGE OF SUM OF BILLED AMOUNT

IV. Following Fee Calculation Methods have been added in “FEE_CALC_METHOD_CD” lookup for “CYCLE BASED COLLECTION LATE FEE”.

a. FLAT AMOUNT
b. PERCENTAGE OF SUM OF PAYMENT DUE
c. PERCENTAGE OF SUM OF STANDARD PAYMENT
d. PERCENTAGE OF SUM OF BILLED AMOUNT
e. PERCENTAGE OF PAYMENT DUE
f. PERCENTAGE OF STANDARD PAYMENT
g. PERCENTAGE OF BILLED AMOUNT
h. PERCENTAGE OF TOTAL DUE AMOUNT
i. PERCENTAGE OF SUM OF TOTAL DUE AMOUNT

V. Similarly, above Fee Calculation Methods have been also added in “FEE_CALC_METHOD_CD” lookup for “CYCLE BASED LATE FEE”.

a. FLAT AMOUNT
b. PERCENTAGE OF SUM OF TOTAL DUE AMOUNT
c. PERCENTAGE OF SUM OF PAYMENT DUE
d. PERCENTAGE OF SUM OF STANDARD PAYMENT

e. PERCENTAGE OF SUM OF BILLED AMOUNT
f. PERCENTAGE OF TOTAL DUE AMOUNT
g. PERCENTAGE OF PAYMENT DUE
h. PERCENTAGE OF STANDARD PAYMENT
i. PERCENTAGE OF BILLED AMOUNT

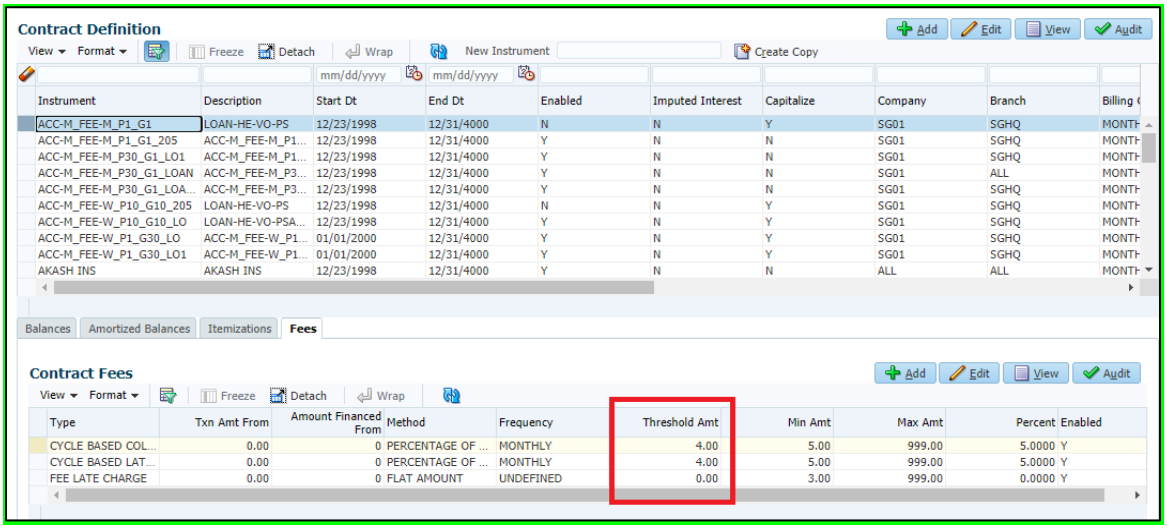
VI. Following Balances have been added for the new Fees under Setup >Contract >Balances for LOAN/LINE LEASE.

Balance	Charge off Method	Write-off Method	Reschedule Method	Billed	Accrued	Non-Performing Rollover	Non-Performing Balance Type
CYCLE BASED COLLECTION LATE FEE	CHGOFF BALANCE	WAIVE	ROLLOVER BALANCE	N	N	N	NONE
CYCLE BASED LATE FEE	CHGOFF BALANCE	WAIVE	ROLLOVER BALANCE	N	N	N	NONE

VII. Following new batch jobs have been introduced under SET-TPE to process “Cycle based Collection Late Fee and Cycle Based Late Fee” respectively.
a. CYCLE BASED COLLECTION LATE FEE PROCESSING
b. CYCLE BASED LATE FEE PROCESSING

VIII. A new field Threshold Amount has been introduced for Fee processing in Setup> Products> Contract > Fees screen.

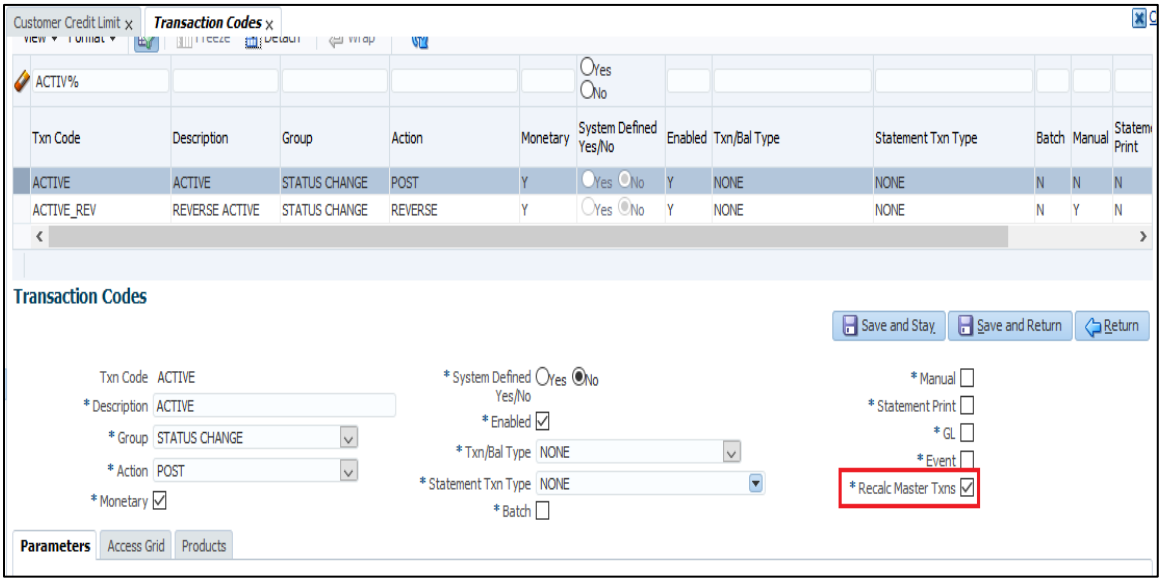
Setup> Products> Contract > Fees



a. This field is applicable and is enabled for input only for new Cycle Based Late Fee and Cycle Based Collection Late Fee.
b. System will validate that Threshold Amt <= Min Amt during Fee setup. Else, otherwise error message: **Threshold Amount cannot be greater than Min amt will be shown on Save**.
c. During Fee calculation, if calculated fee amount is < threshold amount, fee will be posted with transaction amount = 0
d. If calculated fee amount is >= threshold amount, only then existing min-max amt logic will be used for Fee posting.

IX. New field **"Recalc Master Txns"** is added under Setup >System >Transaction Codes to identify monetary transactions that is to trigger recalculation of Late Charge, Cycle Based Late Fee and Cycle Based Collection Late Fee at Master Account level when backdated transaction is posted on any Associated Account that is marked for fee consolidation.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Recalc Master Txns	Y	Check Box	N	Will be enabled for Monetary Transactions only



a. This field is enabled only for Transaction Codes under super group ACCOUNT MONETARY TXN.

b. Default value for this flag is N for all Monetary Transaction codes. i.e. by default, recalculation is disabled for all Monetary Transactions. Based on business requirement, this field can be enabled for required Transaction Codes.

c. If this flag is checked for a Monetary Transaction Code, while initiating any transaction on an associated account using the said Transaction Code, system will validate the following:

- Current transaction is backdated
- Associated account is marked for fee consolidation at Master Account level for any of the fees, i.e. - Cycle Based Late Fee, Cycle Based Collection Late Fee and Late Charge.
- Master account is also be marked for consolidation for the corresponding fees.
- If all the above condition is satisfied, along with original transaction processing on associated account, system will also recalculate and repost the new consolidated fee at Master Account level.
- **Please note no recalculation/reposting will happen on any other Associated Account except the one where the parent transaction was initiated.**

- **Origination Changes**

I. Following new fields have been added to Origination >Contract >Contract (2) under new “Fee Consolidation” header after the Due Date header.

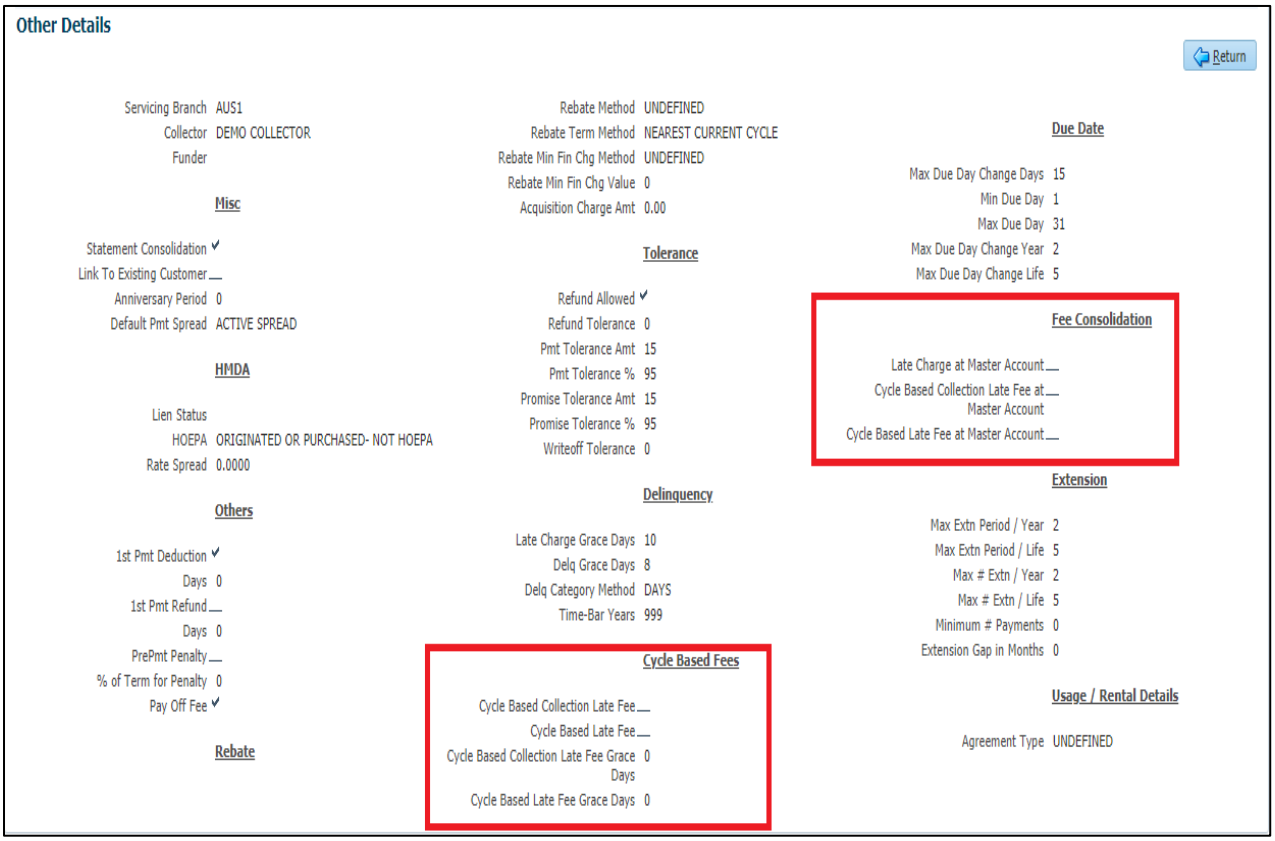
a. These fields are propagated from Setup >Contract when an instrument is loaded and needs to be reloaded if the instrument is changed.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Late Charge at Master Account	Y	Check Box	N	Read Only
Cycle based Collection Late Fee at Master Account	Y	Check Box	N	Read Only
Cycle Based Late Fee at Master Account	Y	Check Box	N	Read Only

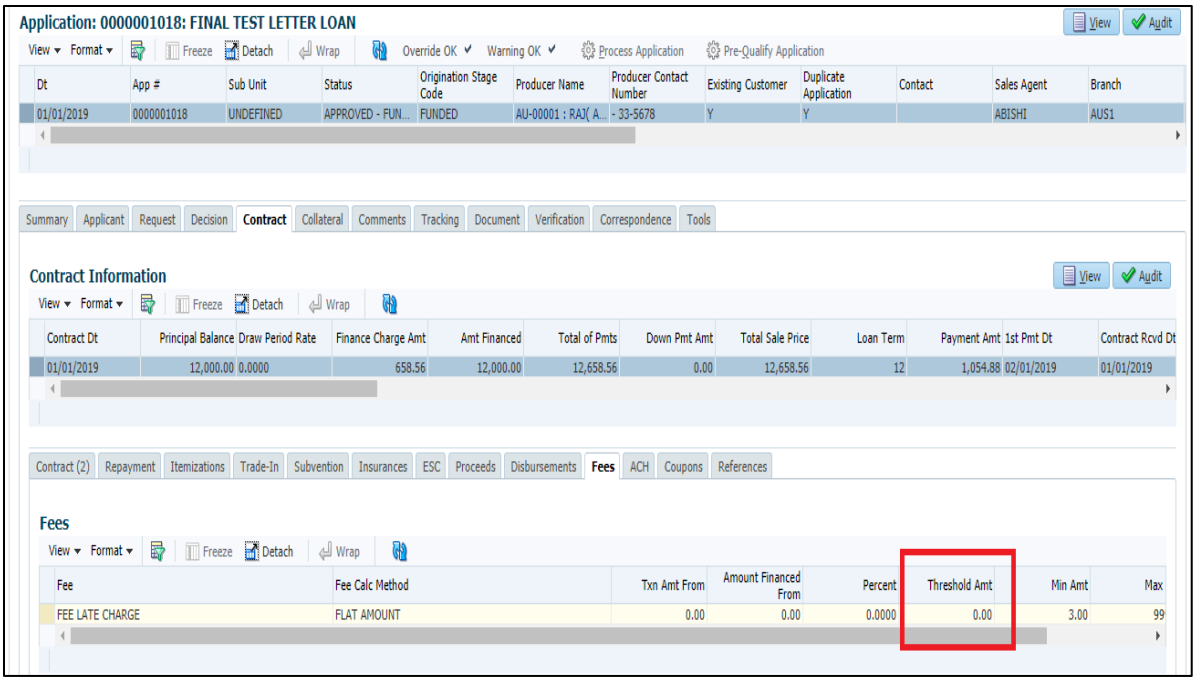
II. Following new fields have been added to Origination >Contract >Contract (2) under “Delinquency” header after Delq Grace Days field.

a. These fields also are be propagated from Setup >Contract when an instrument is loaded and is reloaded if the instrument is changed.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Cycle Based Late Fee	Y	Check Box	N	Read Only
Cycle Based Collection Late Fee	Y	Check Box	N	Read Only
Cycle based Collection Late Fee Grace Days	Y	Text Box (Number)	0	Read Only
Cycle based Late Fee Grace Days	Y	Text Box (Number)	0	Read Only



III. New field Threshold Amt has been added under Origination >Contract > Fees screen



IV. Following Contract Edits have been added to ensure if the Fee configuration is properly configured between Master and Associated Accounts.

a. These Edits have been defined for UI Funding, Conversion and Onboarding.

Description	Default Value	Value	Validation
LATE CHARGE IS NOT CONFIGURED	NA	ERROR	Trigger if the Late Charge is not configured at Associated Application and if the Late Charge Grace Days and/or Billing Cycle is different from Master account, **Applicable in case of "Late Charge at Master Account" is set to "Y" configured at current Application >Contract.**
CYCLE BASED COLLECTION LATE FEE IS NOT CONFIGURED	NA	ERROR	Trigger if the Cycle based Collection Late Fee is not configured at Associated Application and if the Cycle based Collection Fee Due Days, Grace Days, Cycle Base Collection Late Fee Cycle and/or Billing Cycle is different from Master account. **Applicable in case of "Cycle Based Collection Late Fee at Master Account" is set to "Y" and "Cycle Based Collection Late Fee" is set to "Y" at current Application >Contract.**

CYCLE BASED LATE FEE IS NOT CONFIGURED	NA	ERROR	Trigger if the Cycle Based Late Fee is not configured at Associated Application and if the Cycle based late fee Due Days, Grace Days, Cycle Base Late Fee Cycle and/or Billing Cycle is different from Master account, **Applicable in case of "Cycle Based Late Fee at Master Account" is set to "Y" and "CYCLE BASED LATE FEE" is set to "Y" configured at current Application >Contract.**
CYCLE BASED COLLECTION LATE FEE FLAG NOT SELECTED	NA	ERROR	Trigger if Cycle based Collection Late Fee at Master Account flag is checked and if the Cycle based Collection Late Fee flag is unchecked
CYCLE BASED LATE FEE FLAG NOT SELECTED	NA	ERROR	Trigger if the Cycle Based Late Fee at Master Account is Checked and cycle based late fee flag is unchecked
CYCLE BASED LATE FEE CYCLE CANNOT BE GREATER THAN ACCOUNT BILLING CYCLE	NA	ERROR	Trigger if Cycle Based Late fee cycle (e.g. Annual) > Account Billing Cycle (e.g. Monthly). CASE: If Cycle based Late Fee flag is checked.
CYCLE BASED COLLECTION LATE FEE CYCLE CANNOT BE GREATER THAN ACCOUNT BILLING CYCLE	NA	ERROR	Trigger if Cycle Based collection Late fee cycle (e.g. Annual) > Account Billing Cycle (e.g. Monthly) CASE: If Cycle based Collection Late Fee flag is checked.

- **Servicing Changes**

I. Following new fields have been added to Servicing >Account Details >Contract Information >Contract tab under new "Fee Consolidation" header [after the "Delinquency" header].

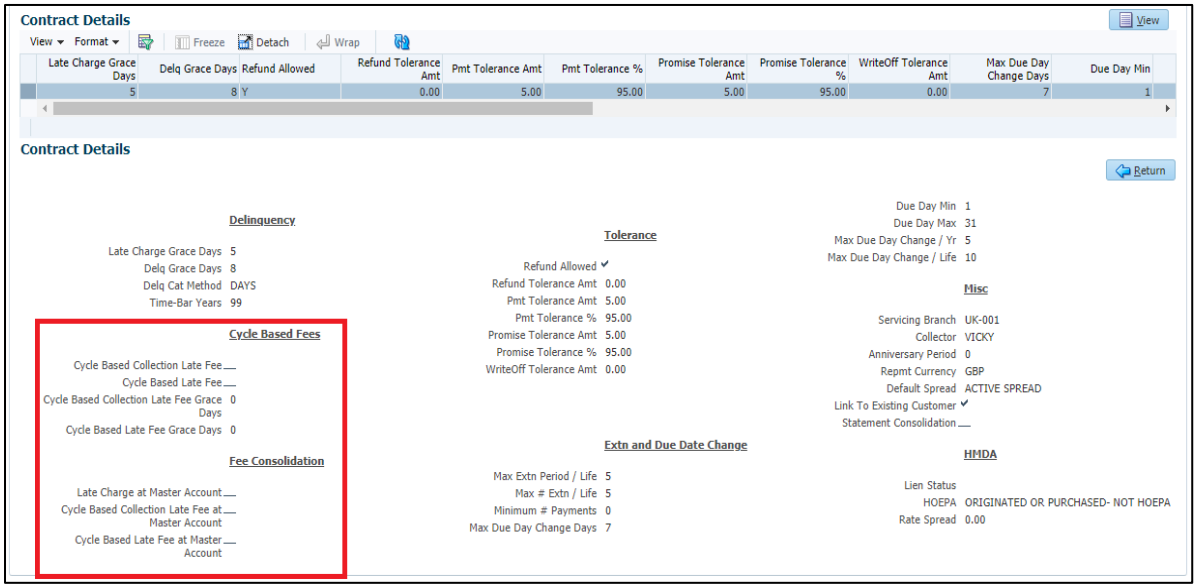
Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Late Charge at Master Account	Y	Check Box	N	Read Only
Cycle based Collection Late Fee at Master Account	Y	Check Box	N	Read Only

Cycle Based Late Fee at Master Account	Y	Check Box	N	Read Only

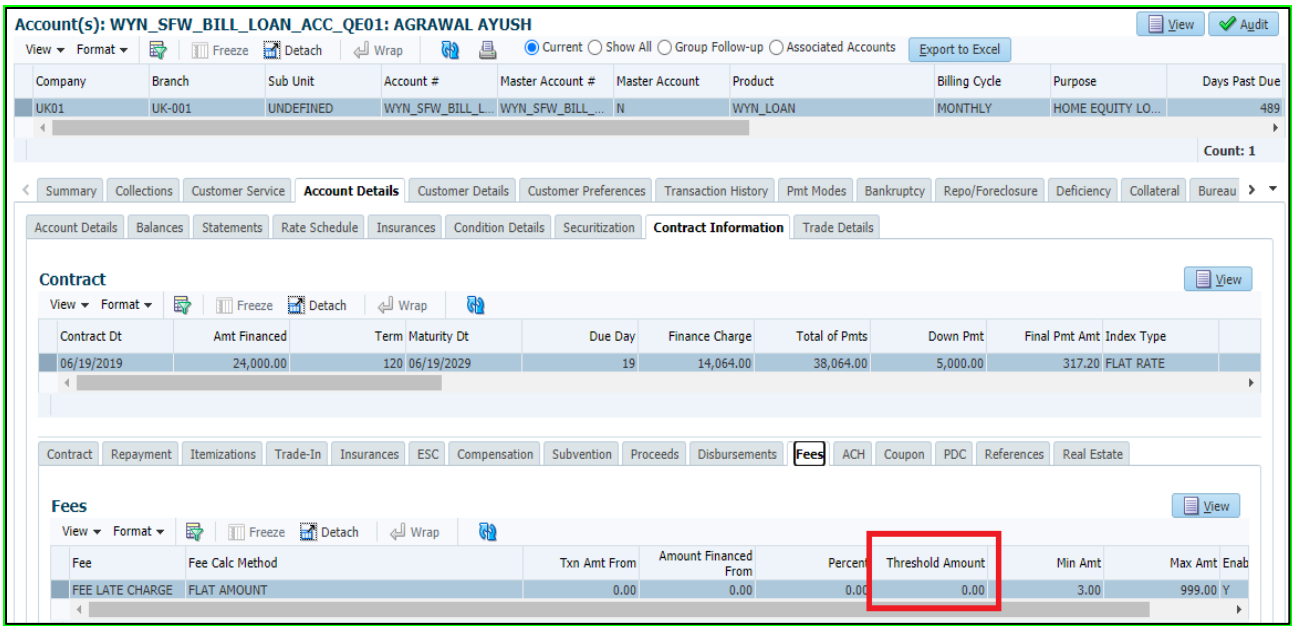
II. Following new fields have been added to Servicing >Contract Information>Contract under “Delinquency” header after Delq Grace Days field.

a. These fields are propagated from Origination >Contract, when an Application is Funded.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Cycle Based Late Fee	Y	Check Box	N	Read Only
Cycle Based Collection Late Fee	Y	Check Box	N	Read Only
Cycle based Collection Late Fee Grace Days	Y	Text Box (Number)	0	Read Only
Cycle based Late Fee Grace Days	Y	Text Box (Number)	0	Read Only



III. Added threshold Amt field in Servicing >Account Details >Contract Information >Fees screen



IV. Following new fields have been added in Servicing >Customer Service >Account Details next to Statement header under new “Fee Consolidation” header.

a. These fields are propagated from Origination as part of funding.

Field Name	Mandatory Y/N	Data Type/ Size	Default Values	Field Validation and Comments
Late Charge at Master Account	Y	Check Box	N	Read Only
Late Charge Grace Days	Y	Text Box (Number)	0	Read Only
Cycle based Collection Late Fee at Master Account	Y	Check Box	N	Read Only
Cycle based Collection Late Fee Grace Days	Y	Text Box (Number)	0	Read Only
Cycle based Late Fee at Master Account	Y	Check Box	N	Read Only
Cycle based Late Fee Grace Days	Y	Text Box (Number)	0	Read Only

V. Existing “MASTER ACCOUNT MAINTENANCE” transaction has been enhanced to validate the below:

a. During addition of an Associated Account to a Master Account which is enabled for Fee Consolidation, system has been enhanced to validate If the corresponding Fee balances are available for Associated Account. Else system will show an error message “<<Balance Name>> not available at Account”.

b. Also, validation has been added to ensure if Account’s Due Days, Billing cycle, Fee Cycle and respective fee grace days are same on associated and master account. Else, system will display error message: “Billing Cycle, Fee cycle and/or <fee name> grace days not same as Master account”. This validation is enforced only when Fee at Master Account flag = Y on Master Account and Fee at Master Account flag = “Y” on Associated account.

VI. Following New Monetary Transaction has been introduced to update the Fee Consolidation Indicator and Grace Days for Associated/Master Account.

Txn Code	Description	Group	Action	Monetary	System Defined	Enabled	Txn/ Bal Type	Stmt Txn Type	Batch	Manual	Stmt Print	GL
FEE_CONSOLIDATION_MAINT	Fee Consolidation Maintenance	ACOUNT MONETARY	POST	Y	N	N	NONE	NONE	N	Y	N	N

Transaction Parameters for above Transaction:

Parameter Name	Type	Default Value	Comments
Txn Date	Date	Sys Date	
Late Charge at Master Account	LOV (Y/N)	N	
Late Charge Grace Days	Number	0	System will update Late Charge Grace Days only if the new value provided is greater than “0”, else existing fields will not be updated.
Cycle based Collection Late Fee at Master Account	LOV (Y/N)	N	

Cycle based Collection Late Fee Grace Days	Number	0	System will update only if the new value provided is greater than “0”, else existing fields will not be updated.
Cycle Based Late Fee at Master Account	LOV (Y/N)	N	
Cycle Based Late Fee Grace Days	Number	0	System will update only if the new value provided is greater than “0”, else existing fields will not be updated.
PROCESS MASTER ASSOCIATED ACCOUNTS	LOV (Y/N)	N	If this parameter is checked, system will post this transaction in all the associated accounts
PROCESS SAME PRODUCT TYPE AND FUNDING TYPE ACCOUNTS	LOV (Y/N)	N	If this parameter is checked, system post this transaction in all the associated accounts which are having the same Product and Funding type as Master Account
VALIDATE PORTFOLIO COMPANY	LOV (Y/N)	N	If checked, validate Current account is having same portfolio Company as Master Account. If not show an error message and will not allow to post the transaction.
REASON CODE	LOV		

a. The above transaction is allowed only for the accounts which are linked to Master Account (Including Master Account), otherwise, system shows an error message “Account not linked to Master Account; Transaction posting not allowed”.

b. If Fee Consolidation at Master Account flags are set to “Yes”, then system validates respective fee balance(s) are available at Master and Associated account or not. If not, then error message is shown - “<<Fee Consolidation>> balances are not available”. Balance validations is being enforced only when both Master and Associated accounts have “Fee at Master Account flag = Y”

c. If Account’s billing Cycle is Monthly, Weekly, Bi-Weekly, Semi-Monthly, and if Late fee Grace Days > 28 then System will throw error message: ‘For billing cycle <<Billing Cycle>> late charge grace days cannot be greater than 28.’

   If Fee at Master Flag is being updated as Y part of this transaction on Master account, then system validates if Associated Account (which will result in fee at master flag = Y) has same respective fee grace days and fee cycle else System throws an error message: ‘Grace Days and Fee cycle of Master and Associated Account must be

II. Existing Billing cycle change transaction has been enhanced to validate the below when posted on Master or Associated account:

a. If fee at master flag is 'Y' on Master account, then

i. Account Billing cycle must result to be greater than or equal to fee cycle.

ii. Account billing cycle must result to be same for Master account and associated account with fee at master flag = 'Y'

b. If fee at master flag is N on Master account, then

i. If fee at respective account(s) flag is 'Y' then, Account Billing cycle must result to be greater than or equal to fee cycle.

ii. If fee at respective account is 'N' then, no validation related to fee cycle will be done.

- **Webservice Changes**
  The newly added fields in the Account level has been included in the Account Details webservice.

- **Batch Job Changes**

  Following matrix outlines the general processing logic for Late Charge, Cycle Based Collection Late Fee and Cycle Based Late Fee batch jobs while based on the newly introduced Fee indicators introduced at account level.

Flag	Master Account (M1)	Associated Account (A1)	Action
Late Charge/Cycle Based Late Fee/Cycle Based Collection Late Fee at Master Account	Y	Y	Looping through Master and Associated Accounts, Posting Late Charge at Master Account and a Zero amount Late Charge transaction at Associated Account.
	Y	N	Looping through Master and Associated Accounts to find out that there are no Associated Account with fee at master flag = Y. Then considers only the Master Account Due Amount to calculate fee According to Fee calculation Method.
	N	N	Not looping through Master and Associated, Posting Late Charge at Master and Associated Based on the Fee Configuration at respective account >Contracts.

	N	Y	Not looping through Master and Associated, Posting Late Charge at Master and Associated Based on the Fee Configuration at respective account >Contracts.

- **Late Charge Processing Batch Job Changes (SET: TPE > TXNLTC_BJ_100_01)**

II. If “Late Charge at Master Account” is selected for Master Account, then system loops through all the associate accounts with “Late Charge at Master Account” = Y and posts Late Charge on the master account as per calculation method define at Master Account level.

   a. This is done even if Master account is not delinquent, but at least one Account among the Master and Associated Account is Delinquent

III. While calculating consolidated Late Charge at Master Account level, only Master Account fee calculation parameters – i.e. Calculation Method, Fee Cycle, Fee Percentage etc. is considered. Associated Account fee parameters do not play any role in scenarios where Late Charge consolidation is enabled at Master Account.

Example: if the Master Account M1 and Associated Accounts A1, A2 and A3 are having the

Late Charge at Master Account = Y.

Late Charge Calculation Method = FLAT AMOUNT

Flat Amount = $ 30

Account	Delinquent	Late Charge at Account
M1	N	30
A1	Y	0
A2	Y	0
A3	N	0

IV. If “Late Charge at Master Account” is Y for an Associated Account and ‘Late charge at Master Account’ is selected for Master Account, then 0 amount is posted on Associated Account.

V. If “Late Charge at Master Account” is selected for an Associated Account but ‘Late charge at Master Account’ is not selected for Master Account, then posting late charge transaction is based on fee configuration at respective account.

Example: If the Master Account M1 and Associated Accounts A1, A2 and A3 are having the

Late Charge at Master Account = Y.

Late Charge Calculation Method = PERCENTAGE OF SUM OF PAYMENT DUE

Percentage = 3%

Account	Delinquent	Payment Due Amount (2/1/2020)	Late Charge at Account on (2/1/2020)
M1	N	0	(0+100+200+0 = 300) X 3% = 9
A1	Y	100	0
A2	Y	200	0
A3	N	0	0

VI. If a backdated Transaction is being posted in Master or Associated Accounts which might result into change in Payment Due Amount, then System recalculates the Late Charge and reposts the Updated Late Charge at Master Account.

VII. When backdated transaction was initiated for Associated Account marked for consolidation, system will check if newly introduced flag **"Recalc Master Txns"** is enabled for the transaction code that initiated the parent transaction in associated account and if so, then system will recalculate and repost the updated Late Charge at Master Account also.

Please note no recalculation/reposting will happen on any other Associated Account except the one where the parent transaction was initiated.

Example: Back Dated Payment Posted on A2 with $ 100 then System reverses the Existing Late Charge Transaction and Repost with updated Late Charge at Master Account.

Account	Delinquent	Back Dated Payment Posted on (1/15/2020)	Payment Due Amount on (2/1/2020)	Late Charge at Account on (2/1/2020)
M1	N		0	(0+100+100+0 = 200) X 3% = 6
A1	Y		100	0
A2	Y	100	(200 – 100 = 100)	0
A3	N		0	0

I. Below new Fee Calculation Methods have been introduced in system for Late Charge processing.

a. PERCENTAGE OF SUM OF PAYMENT DUE AMOUNT is the sum of Due Amount for current due date of Master and associated account (where fee at master flag = Y).

b. PERCENTAGE OF SUM OF BILLED AMOUNT is the sum of Billed Amount for current due date of Master and associated account (where fee at master flag = Y).

c. Similarly, PERCENTAGE OF SUM OF STANDARD PAYMENT is the sum of Standard Payment Amount for current due date of Master and associated account (where fee at master flag = Y).

- **Cycle based Collection Late Fee Processing Batch Job Changes (SET: TPE > New Batch Job)**

I. Batch job will process all Accounts, if Cycle based Collection Late Fee is selected for account and Cycle based Collection Late Fee run date next < GL Date + 1.

II. In case of backdated account creation, system will perform catch-up fee processing, similar to existing behavior for Late fee.

III. If "Cycle based Collection Late Fee at Master Account" is Y for a Master Account, then system loops through all the Associate Accounts having "Cycle based Collection Late Fee at Master Account" as Y and posts Cycle based Collection Late Fee on the master account as per calculation method and Frequency set at Fee definition.

a. This is done even if Master account is not delinquent, but at least one Account among the Master and Associated Account is Delinquent.

b. Fee is posted with a non-zero amount on the Associated/Master account only if due date oldest + fee grace days is less than equal to fee posting date.

IV. While calculating consolidated Cycle Based Collection Late Fee at Master Account level, only Master Account fee calculation parameters – i.e. Calculation Method, Fee Cycle, Fee Percentage etc. is considered. Associated Account fee parameters do not play any role in scenarios where Cycle Based Collection Late Fee consolidation is enabled at Master Account.

Example: If the Master Account M1 and Associated Accounts A1, A2 and A3 are having the

Cycle based Collection Late Fee at Master Account = Y.

Cycle based Collection Late Fee Calculation Method = FLAT AMOUNT

Flat Amount = $ 30

Account	Delinquent	Late Charge at Account
M1	N	30
A1	Y	0
A2	Y	0
A3	N	0

V. If "Cycle Based Collection Late Fee" is Y for an Associated Account and if 'Cycle Based Collection Late Fee' is selected for Master Account, then Cycle based Collection Late Fee is posted with 0 amount on the Associated Account.

VI. If “Cycle Based Collection Late Fee” is Y for an Associated Account but ‘Cycle Based Collection Late Fee’ is not selected for Master Account, then Fee posting happens based on fee configuration at respective account.

Example: if the Master Account M1 and Associated Accounts A1, A2 and A3 are having the

Cycle based Collection Late Fee at Master Account = Y.

Cycle based Collection Late Fee Calculation Method = PERCENTAGE OF SUM OF PAYMENT DUE Percentage = 3%

Account	Delinquent	Payment Due Amount	Late Charge at Account
M1	N	0	(0+100+200+0 = 300) X 3% = 9
A1	Y	100	0
A2	Y	200	0
A3	N	0	0

VII. If there is a backdated Transaction posted in Master or Associated Accounts which might result into change in Payment Due Amount, System recalculates the Cycle based Collection Late Fee and repost the Updated Cycle based Collection Late Fee at Master Account.

VIII. When backdated transaction was initiated for Associated Account marked for consolidation, system will check if newly introduced flag **“Recalc Master Txns”** is enabled for the transaction code that initiated the parent transaction in associated account and if so, then system will recalculate and repost the updated Cycle based Collection Late Fee at Master Account also.

Please note no recalculation/reposting will happen on any other Associated Account except the one where the parent transaction was initiated.

Account	Delinquent	Back Dated Payment Posted on (1/15/2020)	Payment Due Amount on (2/1/2020)	Late Charge at Account
M1	N		0	(0+100+100+0 = 200) X 3% = 6
A1	Y		100	0
A2	Y	100	(200 – 100 = 100)	0

A3	N		0	0

IX. If Cycle Based Collection Late Fee at Master = Yes, then below processing logic is followed

a. System arrives at initial Run Date Next = First Due Date + Grace Days
b. Post first run date, system calculates next run date based on fee frequency.
c. Due Amount is being considered only when the Due date + Grace Days is <= Run date Next, if it's greater than Due Date then respective due amount is not considered for Fee calculation.

X. Below new Fee Calculation Methods have been introduced in system for Cycle Based Collection Late Fee and Cycle Based Late Fee processing.

a. PERCENTAGE OF SUM OF TOTAL DUE AMOUNT is the sum of 'Total Due' of Master and associated account (where fee at master flag = Y).
b. PERCENTAGE OF TOTAL DUE AMOUNT is the 'Total Due' of one Account
c. If Consolidate at Master flag is N but fee Calculation Method is "PERCENTAGE OF SUM OF TOTAL DUE AMOUNT" then it behaves just like "PERCENTAGE OF TOTAL DUE" as there is only single account involved.
d. PERCENTAGE OF SUM OF PAYMENT DUE AMOUNT is the sum of Due Amount for current due date of Master and associated account (where fee at master flag = Y).
e. PERCENTAGE OF SUM OF BILLED AMOUNT is the sum of Billed Amount for current due date of Master and associated account (where fee at master flag = Y).
f. Similarly, PERCENTAGE OF SUM OF STANDARD PAYMENT is the sum of Standard Payment Amount for current due date of Master and associated account (where fee at master flag = Y).

XI. Below are some examples for cycle-based collection late fee calculation scenarios:

Cycle_Based_Collection_Late_Fee_v1.0.xl

- **Cycle Based Late Fee Processing Batch Job Changes (SET: TPE > New Batch Job)**

i. Cycle Based Late Fee Processing has similar processing logic as Cycle Based Collection Late Fee explained above. Only Cycle Based Late Fee related fields are considered for processing.
ii. Batch job will process all Accounts, if Cycle based Late Fee is selected for account and Cycle based Late Fee run date next < GL Date + 1.
iii. In case of backdated account creation, system will perform catch-up fee processing, similar to existing behavior for Late fee.

iv. If “Cycle based Late Fee at Master Account” is Y for a Master Account, then system loops through all the Associate Accounts having “Cycle based Late Fee at Master Account” as Y and posts Cycle based Late Fee on the master account as per calculation method and Frequency set at Fee definition.

   a. This is done even if Master account is not delinquent, but at least one Account among the Master and Associated Account is Delinquent.
   b. Fee is posted with a non-zero amount on the Associated/Master account only if due date oldest + fee grace days is less than equal to fee posting date.

v. While calculating consolidated Cycle Based Late Fee at Master Account level, only Master Account fee calculation parameters – i.e. Calculation Method, Fee Cycle, Fee Percentage etc. is considered. Associated Account fee parameters do not play any role in scenarios where Cycle Based Late Fee consolidation is enabled at Master Account.

Example: If the Master Account M1 and Associated Accounts A1, A2 and A3 are having the Cycle based Late Fee at Master Account = Y.

Cycle based Late Fee Calculation Method = FLAT AMOUNT

Flat Amount = $ 30

Account	Delinquent	Late Charge at Account
M1	N	30
A1	Y	0
A2	Y	0
A3	N	0

vi. If “Cycle Based Late Fee” is Y for an Associated Account and if ‘Cycle Based Late Fee’ is selected for Master Account, then Cycle based Late Fee is posted with 0 amount on the Associated Account.

vii. If “Cycle Based Late Fee” is Y for an Associated Account but ‘Cycle Based Late Fee’ is not selected for Master Account, then Fee posting happens based on fee configuration at respective account.

Ex: if the Master Account M1 and Associated Accounts A1, A2 and A3 are having the Cycle based Late Fee at Master Account = Y.

Cycle based Late Fee Calculation Method = PERCENTAGE OF SUM OF PAYMENT DUE

Percentage = 3%

Account	Delinquent	Payment Due Amount	Late Charge at Account
M1	N	0	(0+100+200+0 = 300) X 3% = 9
A1	Y	100	0
A2	Y	200	0

A3	N	0	0

viii. If there is a backdated Transaction posted in Master or Associated Accounts which might result into change in Payment Due Amount, System recalculates the Cycle based Late Fee and repost the Updated Cycle based Late Fee at Master Account.

ix. When backdated transaction was initiated for Associated Account marked for consolidation, system will check if newly introduced flag “Recalc Master Txns” is enabled for the transaction code that initiated the parent transaction in associated account and if so, then system will recalculate and repost the updated Cycle based Late Fee at Master Account also.

Account	Delinquent	Back Dated Payment Posted on (1/15/2020)	Payment Due Amount on (2/1/2020)	Late Charge at Account
M1	N		0	(0+100+100+0 = 200) X 3% = 6
A1	Y		100	0
A2	Y	100	(200 – 100 = 100)	0
A3	N		0	0

x. If Cycle Based Late Fee at Master = Yes, then below processing logic is followed

c. System arrives at initial Run Date Next = First Due Date + Grace Days

d. Post first run date, system calculates next run date based on fee frequency.

e. Due Amount is being considered only when the Due date + Grace Days is <= Run date Next, if it’s greater than Due Date then respective due amount is not considered for Fee calculation.

xi. Below new Fee Calculation Methods have been introduced in system for Cycle Based Late Fee and Cycle Based Late Fee processing.

f. PERCENTAGE OF SUM OF TOTAL DUE AMOUNT is the sum of ‘Total Due’ of Master and associated account (where fee at master flag = Y).

g. PERCENTAGE OF TOTAL DUE AMOUNT is the ‘Total Due’ of one Account

h. If Consolidate at Master flag is N but fee Calculation Method is “PERCENTAGE OF SUM OF TOTAL DUE AMOUNT” then it behaves just like “PERCENTAGE OF TOTAL DUE” as there is only single account involved.

i. PERCENTAGE OF SUM OF PAYMENT DUE AMOUNT is the sum of Due Amount for current due date of Master and associated account (where fee at master flag = Y).

j. PERCENTAGE OF SUM OF BILLED AMOUNT is the sum of Billed Amount for current due date of Master and associated account (where fee at master flag = Y).

k. Similarly, PERCENTAGE OF SUM OF STANDARD PAYMENT is the sum of Standard Payment Amount for current due date of Master and associated account (where fee at master flag = Y).

XII. Below are some examples for cycle-based late fee calculation scenarios:

2.18.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.19 Web Service Enhancements

2.19.1 Overview

As part of multiple enhancements, existing web services have been enhanced to handle the changes / impact or new web services created. Check swagger for more information.

Also, clean-up of rec-types done to bring in sync of Post/PUT WS.

2.19.2 Description

New Web Services

Service Name	Purpose / Impact
Work Order Web Services (Post, Put, Get)	New service to create, update and get details of Work Order.
Get Stipulations web service	This service is used to get the list of stipulations that are linked to an application using application number. If the stipulations were not loaded by user, system will load configured stipulations from setup and send it in response.
Setup: Pricing Web Service	This service is used to send the Setup > Pricing > Loan/Line/Lease > Result details as part of the response
Decision Web Service –Get	This service is used to get decision data with multi offer details of an application from OFSLL.
Decision Web Service – Put	This service is used to update the Current decision details of an application.
Simple Application Entry	Using this service, user can create the application with minimum payload information - System will use the available fields data in user defined default setup to populate the application fields based on the user code / authentication user in the request. I.e., system find the records which are matching with user's Organization and Division in the setup - Few of the missing mandatory fields will be defaulted

Changes to existing web Services

Service Name	Purpose / Impact	Feature
Application Search with Status/New Status	Ability to search an application with 'Status' and 'Sub-Status'.	
Application Details web-service **(GET – applicationDetails)**	Added → Payment Amount and Contract Details	
Account Details web-service **(GET – getAccountDetail)**	Added → Billing cycle, Company code, Balance paid and remaining outstanding balance for each balance type	

2.19.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.20 Application Search Web Service - Enhancement

2.20.1 Overview

Ability to search an application with ‘Status’ and ‘Sub-Status’. [Origination]

2.20.2 Description

- Added two new following non-mandatory request parameters to ‘Application Search’ web service.
  - Status
  - Sub-Status

Validation:

- If the invalid 'Status and Sub-Status' are provided in the request, system does not pull the application details and shows an error message as "Please provide valid 'Status and Sub-Status”.

2.20.3 Seed Data

Refer ‘Appendix: Seed Data’ chapter.

2.21 Decision Web Service - GET

2.21.1 Overview

Created new Decision GET web service for OFSLL. This provides ability to get Decision details from OFSLL through web service. [Origination > Underwriting > Decision]

2.21.2 Description

- Created new Decision GET web service, to get decision data with multi offer details of an application from OFSLL.
- Application Number is given as input. If Current Decision exists on the Application, system returns the Decision Details including all the Decisions (Current and Not Current).
- This will be Multi block array. Result varies based on Loan/Line and Lease (Refer to Decision Screen)
- If there is no Current Decision exists on the application, system does 'Select Pricing' and sends the Result.
- If Multi Offer flag is enabled at the Company level system parameter, then system sends the Multiple Offers as a result. Refer to the Multi offer tab [Only for LOAN product].
- All the Offers will be fetched.

Validations:

- If no pricing is matched, then system returns 'No Pricing Matched'.
- If entered application number is wrong, system prompts for error message 'Please enter Valid application number'.
- Application status is to be in 'New-Review Request' status or any next statuses.
- If Application status is in any previous statuses to 'New-Review Request' status then, system prompts error message 'Application status is to be 'New-Review Request' status or any next statuses'.
- If the Application is at any status of "REHASH" then system mandatorily does Select Pricing and then sends the response.

2.21.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.22 Decision Web Service - PUT

2.22.1 Overview

- Created new Decision PUT web service for OFSLL. Third Party portals are allowed to get the Decision and able to update the decision details and send back to the OFSLL. [Origination > Underwriting > Decision]

- In most of the cases dealers will be able to change the rates based on the negotiation with the customer and send back the decision. This scenario will not come into picture if the Application is Auto Decisioned.

2.22.2 Description

- Created new Decision PUT web service, to update the **Current** decision details of an application.
- This service updates **only** Current Decision record on that Application and returns the Decision Details.
- Result varies based on Loan/Line and Lease (Refer to Decision Screen)

Validations:

- If entered application number is wrong, system prompts for error message 'Please enter Valid application number'.
- This service updates only current Decision. Application has Current Decision.
- If there is no Current Decision exists on the application, system prompts error "No Current Decision available for this application"
- Application status has to be in '**New-Review Request**' only.
- If Application status is in any other statuses, system should prompt error message 'Please check Application status should be 'New-Review Request'.
- System validates that "Application Number, Current Indicator and Pricing Code" are related, else prompts an error message as "Application Number, Current Indicator and Pricing Code should be related".

2.22.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.23 Asset File Upload Enhancement

2.23.1 Overview

Bulk file upload feature provided for CLOB = Y, for uploading Asset Details in OFSLL using Input Data Dump files.

2.23.2 Description

Asset file uploads will work for the system parameter "CMN_FILE_PROCESS_TO_LOB"='Y'.

- Asset upload
- Asset Tracking attributes
- Asset Attributes
- Asset valuation

Insert, Update features are provided using the Bulk file upload.

- Files with data from ALL Companies

Ex: For Assets file upload

Keep the files in the folder "/input/respective asset files upload folder"

Select the “ALL” company and run the relevant Batch Jobs in "SET-IFP".

- Files with Specific Company

Keep the files in the folder "/input/respective asset file upload folder/ Company"

Ex: US01 for Assets file upload

Keep the files in the "/input/ias/US01"

Run the "US01" batch job "SET-IFP" and relevant Job

- Below are the list of batch jobs and folders created newly for assets file uploads.

SET-IFP: IASPRC_BJ_100_01: Assets file upload (Folder input/ias).

SET-IFP: IATPRC_BJ_100_01: Asset tracking attributes file upload (Folder input/iat).

SET-IFP: IAVPRC_BJ_100_01: Asset valuations file upload (Folder input/iav).

SET-IFP: IAAPRC_BJ_100_01: Asset attributes file upload (Folder input/iaa).

2.23.3 Seed Data

Refer ‘Appendix: Seed Data’ chapter.

2.24 Bulk GL file processing

2.24.1 Overview

GL bulk file upload feature is enhanced, to be controlled by CLOB parameter.

2.24.2 Description

Bulk file upload feature is enhanced to be controlled by CLOB parameter, this gives a feature to upload input files using the Incoming process files web service. Earlier user expected to keep these files in Specific server path. This enhancement will be used for SaaS customers.

- Bulk updates was previously not handled using CLOB Yes/No.
- Company information are taken from batch job while inserting data.

FOLDER NAME	BATCH NAME	BATCH DESC	FILE_TYPE_CD	FILE TYPE DESC
GAT	BLKGLS_BJ_100_01	BULK UPLOAD FOR GL ATTRIBUTES	GAT	GL ATTRIBUTES
GTR	BLKGLS_BJ_100_02	BULK UPLOAD FOR GL TRANSLATION DEFINITION	GTR	GL TRANSLATION DEFINITION
GTD	BLKGLS_BJ_100_03	BULK UPLOAD FOR GL TRANSACTION TYPES DETAILS	GTD	GL TRANSACTION TYPES DETAILS
GTN	BLKGLS_BJ_100_04	BULK UPLOAD FOR GL TRANSACTION LINKS	GTN	GL TRANSACTION LINKS
PRP	BLKPRP_BJ_100_01	BULK UPLOAD FOR PRICING SETUP	PRP	PRICING SETUP

Ex: For Bulk GL file upload

Keep the files in the folder "/input/respective files upload folder"

Select the specific company and run the relevant Batch Jobs in "SET-BLK".

2.24.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

2.25 Machine Learning for Servicing Queue Creation

2.25.1 Overview

OFSLL is leveraging the Oracle Data Mining Capability to give additional Machine Learning features. Oracle Data Mining provides a powerful, state-of-the-art data mining capability within Oracle Database. Machine Learning capability is leveraged to identify the Queue/Segmentation that can be created for the Account data. Intelligent Segmentation feature will give the list of Clusters/Queues, as the Machine Learning Algorithm is created for a given Condition. This Screen is developed using the Oracle JavaScript Extension Toolkit (JET) frame work to deploy please read the installation manual.

Please refer the Installation document to enable the Machine Learning Capability of Oracle Database in OFSLL.

Orthogonal Partitioning Clustering (O-Cluster), an Oracle-proprietary Clustering algorithm, has been used to create Intelligent Segments/Clusters for a given condition. Clustering Algorithm will discover natural groupings in given data.

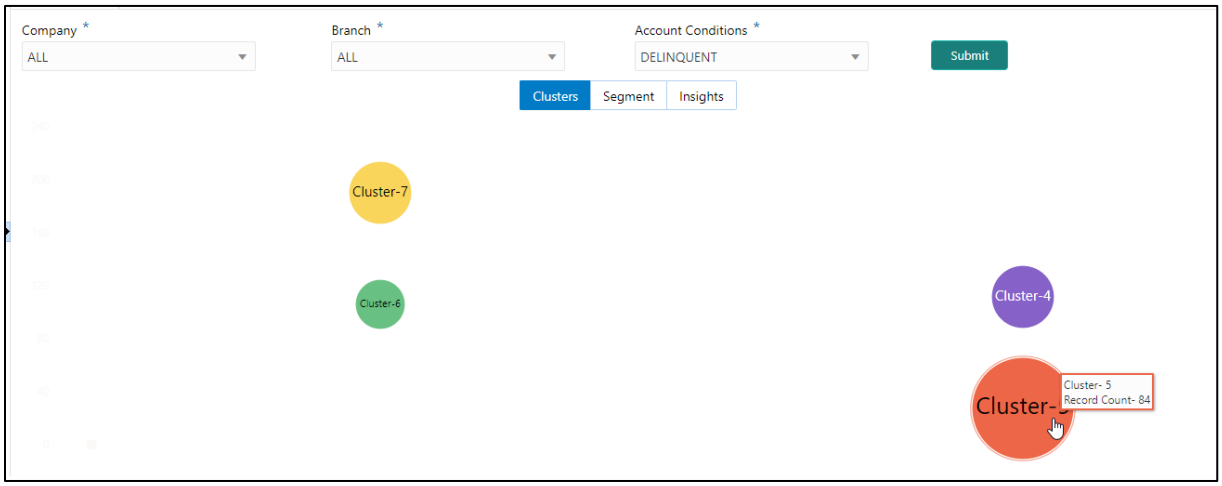
2.25.2 Description

New Setup Menu is introduced in User “Intelligent Segmentation”. User will have option to select the Company, Branch and Account Condition.

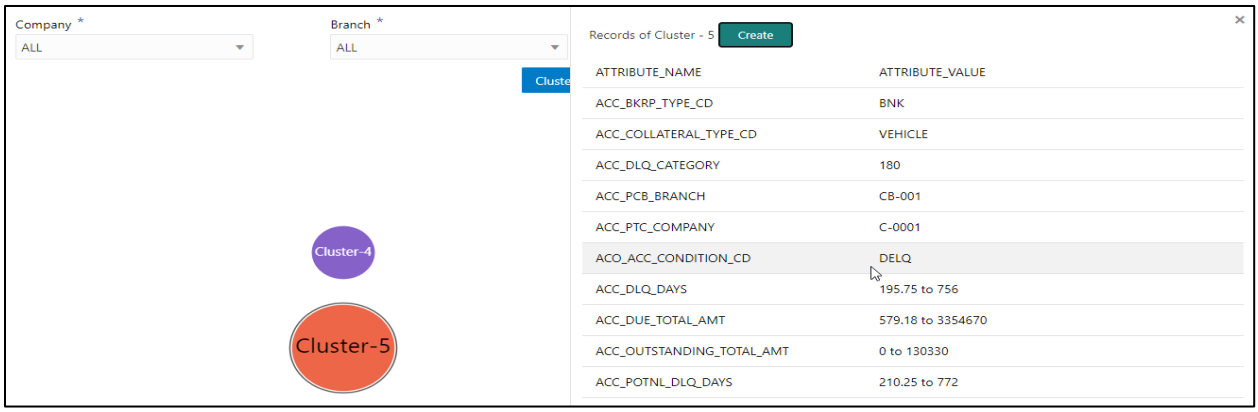
When user “Submit” System will create the Clusters and display on the below user Interface. Screen will display the record count for each Cluster ID. Size of Clusters on screen will change according to the size of record counts.

Create Button:

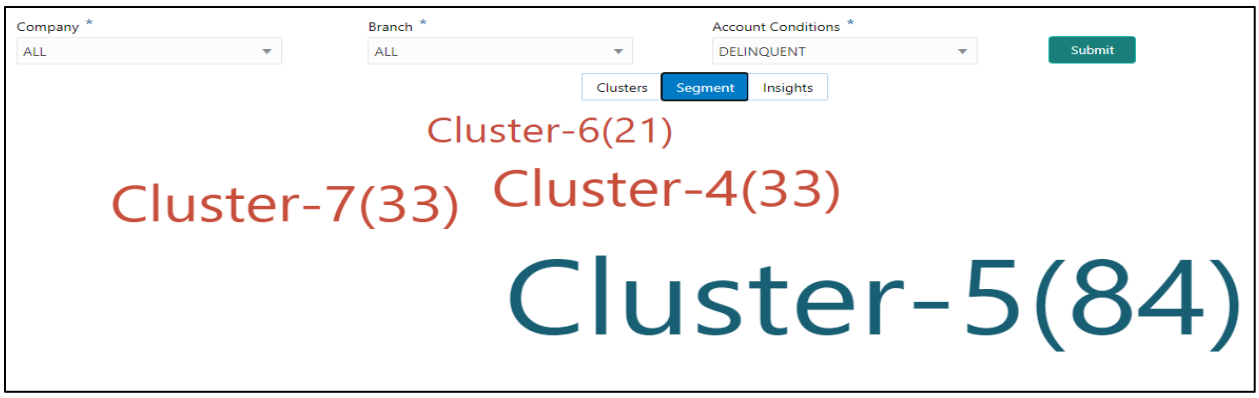
Click on create button will Create the Queue in OFSLL in Disabled status and selection criteria of the Queue will be Populated with the Cluster Attributes.



Click on a Cluster and system display’s the criteria for the Cluster Creation. Oracle clustering algorithm uses attributes to create the cluster, these attributes can be viewed by clicking on the clusters. This will give more information about attributes name and attributes values for given Cluster ID. Each Cluster ID will have its own set of rules.



Segment:
Will Display another visual representation of the Clusters. Screen will display the record count for each Cluster ID.

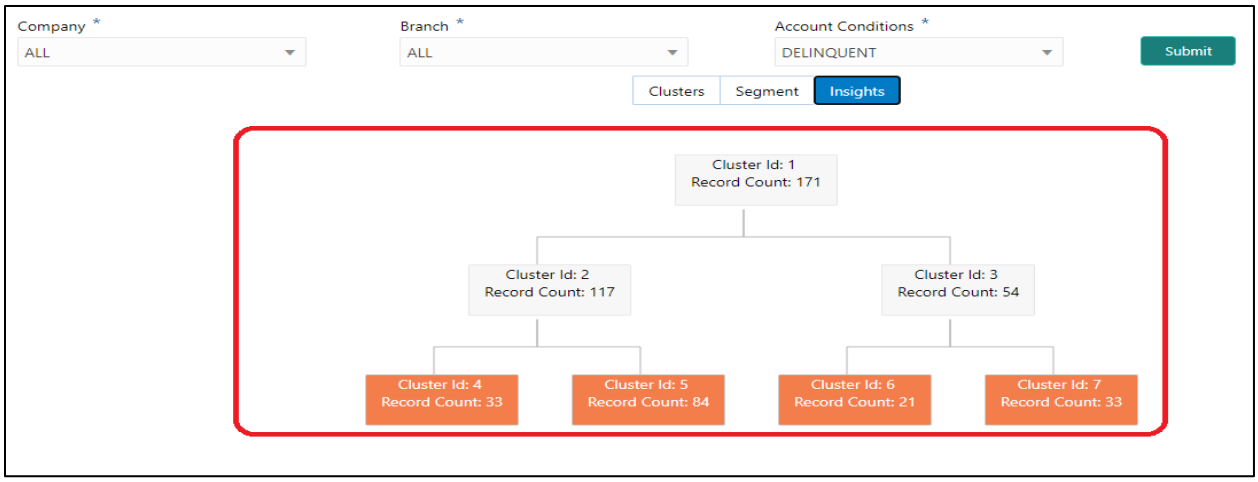


Select a cluster and system gives capability to create the Cluster as Queue with the Criteria.

Note:

- Machine Learning attributes directly inserted as queue criteria and hence user is expected to update the selection criteria in the queue and ensure the formula is formatted well to enable the queue.
- To utilize the Oracle ML capability, financial institute is recommended to run this on non-production / data warehouse database.

Insights:
Another visual representation of the Clusters identified by the Machine Learning. The clustering algorithms supported by Oracle Data Mining perform hierarchical clustering. Before arriving at the final cluster it divides the data in various intermediary clusters. The leaf clusters are the final clusters generated by the algorithm.



Select a cluster and system gives capability to Create the Cluster as Queue with the Criteria.

2.25.3 Seed Data

Refer 'Appendix: Seed Data' chapter.

3. Appendix: Seed Data

Seed data for all the impacted modules against specific enhancement is available in the below attached document.

Seed Data_14.10.0.0.0_v1.2

To view, you can either click on the icon or open from attached documents in left menu.

4. Patches and Bugs

Bug Id	Bug Description	Fix Description
NA	NA	NA

5. Security Fixes

-NA-

6. Limitations and Open issues

Bug	Description
32220126	OJET SCREEN - PROMOTION DETAILS NOT SHOWING IN THE CONTRACT
32216106	OJET-ACCOUNT-SYSTEM THROWS EXPIRATION WARNING EVEN WHEN THE SCREEN IS NOT IDLE
32215494	APPLICATION ENTRY : ISSUE FOR MANDATORY FILEDS IN COLLECTION EXTN > ACCOUNT SEARCH > APPLICATION ENTRY
32220270	OJET-ACCOUNT-NO VALIDATION FOR THE ZIP CODE AND CITY FILEDS
32216828	OJET-ACCOUNT-TELECOM FIELDS SHOULD BE NON MANDATORY
32216597	ADDONS ARE NOT SAVING IN THE COLLATERAL IN OJET ACCOUNT CREATION > COLLATERAL CREEN
32175872	DATA'S ARE NOT GETTING POPULATED FOR LETTERS WITH COMPANY ALL
31287419	INSURANCE EDITS ARE NOT TRIGGERING DURING ACCOUNT CONVERSION FOR LEASE
30554296	CALCULATE PAYMENT WS RETURNING THE AMOUNT FINANCE AS PAYMENT AMOUNT

7. Components of the Software

7.1 Documents accompanying the software

The various documents accompanying the software are as follows:

- Release Notes
- Installation Guides
- Installer Kit
- User Manuals and Installation manuals - These can be accessed from the link https://docs.oracle.com/cd/F35490_01/index.htm

7.2 Software Components

Software Components of this patch release are as follows:

- Core
  - UI Components Ear file (JSF, XML, XLF, JSFF)
  - Stored Procedures (Packages, Views, Java Stored procedures)
  - Reporting Components(Data models(xdm), Reports(xdo, rtf))
  - BIP / canned reports
- Interface
  - Stored Procedures (Packages, Views, Types)
  - The WSDL files for the service supported
  - XSD Structure (dictionary) for the web service
  - Configuration files for the web service
  - Java classes for the web service
  - The service documents – describing the services
  - Extensibility Document – Describes customization for the services.
- Installation utilities
  - Script based installation for Database components
  - Installation documents for Database, UI, Web services

8. Annexure – A: Environment Details

Component	Deployment option	Machine	Operating System	Software	Version
Oracle Financial Services Lending and Leasing	Centralized	Application server	Oracle Enterprise Linux 6.7+ & 7.0 (64 Bit) and Sun SPARC with Oracle Solaris 11 (64 Bit)	Oracle WebLogic Enterprise Edition(Fusion Middleware Infrastructure installer – includes ADF and RCU)	12.2.1.4.0
				Oracle JDK	1.8.0_261
				Application Development Framework	12.2.1.4.0
		Database Server		Oracle Database Enterprise Edition	19.3.0.0.0
		Reporting Server		Oracle Analytics Publisher	5.5.0
		Client Machines	Windows 10 (Patch Version 1703)	Microsoft Edge (64 Bit)	44.18362.449.0
				Mozilla Firefox (32 Bit)	78.5.0esr
				Google Chrome (64 Bit)	87.0.4280.88
			Mac OS X	Apple Safari (64 Bit)	13.1

9. Annexure – B: Third Party Software Details

For information on Third Party Software Details, refer to Licensing guide available in OTN library – https://docs.oracle.com/cd/F35490_01/pdf/refdocs/Licensing_Guide.pdf

ORACLE
Financial Services

Patch Release Notes
Oracle Financial Services Lending and Leasing Release 14.10.0.0.0
September 2025

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
https://www.oracle.com/industries/financial-services/index.html