1 Merchant Place Cnr Fredman Dr & Rivonia Rd Sandton 2196
PO Box 786273 Sandton 2146 South Africa
Switchboard +27 11 282-8000
Facsimile +27 11 282-8008
www.rmb.co.za

RAND MERCHANT BANK
A division of FirstRand Bank Limited

Murex Phase 1B Funding Solution

Project nXt

February 2017

TABLE OF CONTENTS

DOCUMENT CONTROL

Document Type	
Title:	Murex Phase 1B Funding Solution
Submitted to:	Project nXt Knowledge Partners
Created by:	Project nXt Team
Creation Date:	16 February 2017

Document History

Version	Date	Status	Amended By	Comments
V0.1	16/02/2017	First Draft	Meilan Matabadal	
V0.2	02/03/2017		Meilan Matabadal	Updated on Feedback
V0.3	13/03/2017		Meilan Matabadal	Inclusion of funding solution for London

1. INTRODUCTION

The implementation of the front to back to risk architecture in Murex necessitated the assessment of Murex as a replacement for Chameleon for the purposes of funding. Prior to the selection of Murex, as the platform of choice for this architecture, an assessment into RMBs' FTP model, more specifically the operational processes, was executed by Elixirr – a 3rd party consulting firm. The Elixirr assessment concluded that the current CBF model is not commonly used across investment banks and is not an optimal mechanism to affect FTP. Some of the key recommendations were to:

- Move from a reactive, settlement day, cash approach to a predictive cash flow model supported through a funding aggregation solution allowing the ability to view the full forward position profile.
- Transition from a reactive assignment of FTP components to enable for "all-in" charge, including liquidity premiums etc., to be allocated based on the duration and risk of positions.
- Matched funding to be auto-generated between the BU's and GTSY on a transaction level basis where appropriate.
- Further rationalise the entries in the General Ledger by adopting an assumed perfect settlement funding approach. This can be achieved provided up stream controls are in place, which ensure booked cashflows are reliably similar to the settled cashflows and ultimately inform what GTSY must fund.
- Redefine the cash settlement entries to a Nostro/Vostro type model with RMB seeing FNB as their external bank and FNB seeing RMB as a corporate client in order to standardise and simplify the settlement process

Subsequently the Murex capability was assessed at a high level against these recommendations and deemed a suitable platform to replace Chameleon both from a system and as a means to migrate away from the cash based funding approach embedded in the bank.

1.1 Murex Phase 1B: Interim Solution

During the high level assessment of Murex's funding solution; the following key issues were unpacked in relation to the implementation.
Murex has two mechanisms to effect funding "Theoretical Funding" and "Real Funding".

- Theoretical funding dynamically generates the funding or financing costs applicable to transactions whilst the transaction is revalued. The financing cost is based on the overnight rate set per unique closing entity defined. Reporting is then used to report the funding of a portfolio or transaction.
- Real funding is executed through the generation transactions (typically loans) between a portfolio and its designated funding desk. Funding desks can be created, at one or more various points in the portfolio hierarchy before finally being swept to GTSY. This process would require the implementation of loans, which is only planned for phase 2.

Both solutions are a significant departure from the existing process, and would require significant changes in both operational, control fabric and finance processes. Switching all of the banks funding process to either approach requires significant time to assess impact to the existing environment. Enabling either process only for trades executed on Murex, would require 2 operating models for funding at least for Phase 1B of nXt and this was expected to add unnecessary risk to an already critical process. Phase 1B is limited to Fixed Income and IRD products that will be migrated onto the Murex platform.
Further, detailed impact analysis across the operations, finance and GTSY processes is required to determine the change required to roll out either Murex solution.

The banks currency funding is currently managed through FX transactions. FX as an asset class is only scheduled for Phase 2.

Given the above considerations and further impact analysis required, it was determined to keep Chameleon as the funding platform for phase 1B. The details of how Murex will integrate into the banks existing funding processes is detailed below.

The analysis of the change impact of Murex funding, will be initiated for detailed analysis, in parallel to the Phase 1 implementation.

1.2 Business Context

Calypso currently serves as the bank's settlement engine. All trading platforms are integrated with Calypso to ensure that every theoretical cashflow recorded on a trade is physically settled.

RMB employs cash base funding methodology where every cashflow recorded on a trade is funded via a funding account against Group Treasury. Each portfolio and currency combination has its own treasury funding account (TS account), which attracts daily interest and is recorded in the ledger. These funding accounts ensure each portfolio has a balanced balance sheet. The balances and interest calculations on these funding accounts are housed in Chameleon.

To enable an update on a funding account, Calypso maintains a bidirectional relationship with Chameleon. As trades are settled in Calypso; real-time funding requests are sent to Chameleon to update the funding account. Upon receipt of the funding request, Chameleon sends an acknowledgement message back to Calypso.

Business events are generated at the end of the day to reflect the aforementioned process for financial and other reporting. These include:

- Cashflow event – reflects the theoretical settlement, which is sent through on settlement date from the source system
- Payment – generated once the physical payment has occurred from Calypso
- Funding events – generated on receipt of the funding request . Chameleon generates two funding events as part of the overnight process. The first reflects the movement on the trading book and the second shows the corresponding movement in treasury.

As part of Phase 1B, Murex will function as the settlement engine for all trades captured in the system. This would require that the Murex conform to the current funding model by sending funding requests to Chameleon and generating business events to meet downstream reporting requirements. This aligns to the existing settlement process adopted by Calypso.

2. PHASE 1B FUNDING SOLUTION

2.1 Scope

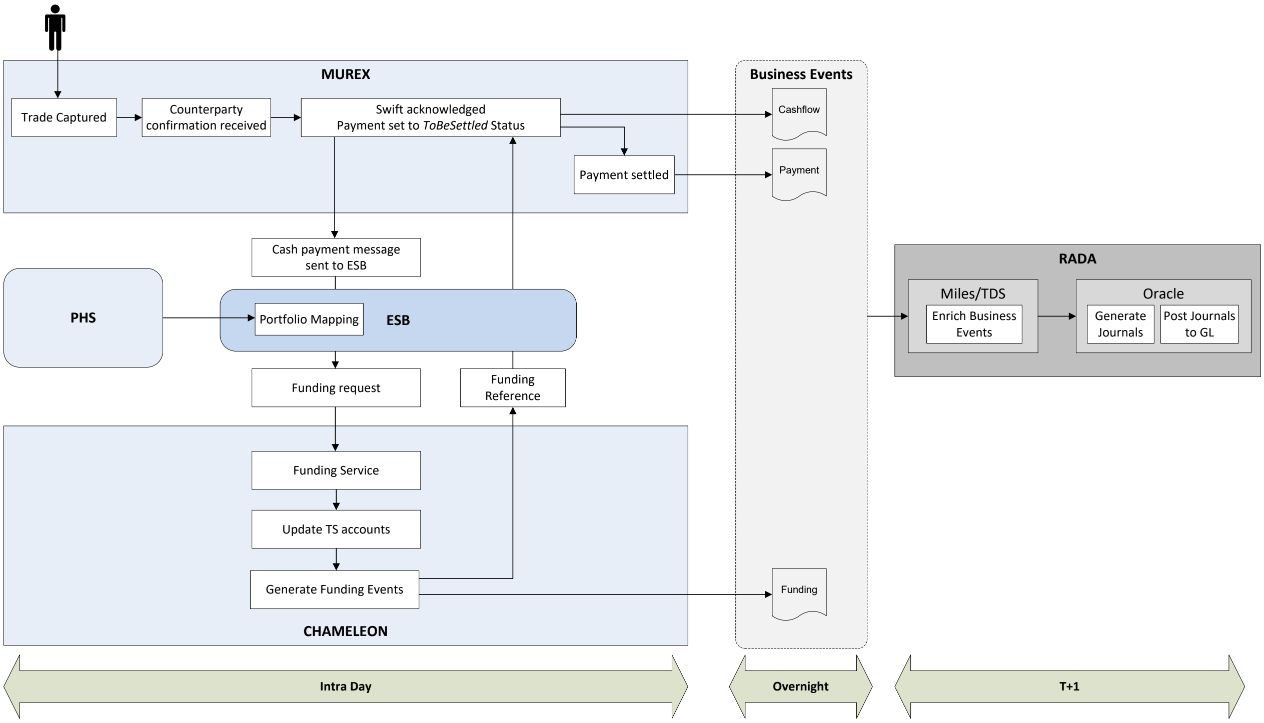
This solution will apply to all internal and external trades captured in Murex as part of Phase 1B. Phase 1B includes all trades booked on portfolios that contain Fixed Income and IRD products. In the event a portfolio contains trades pertaining to other products, in addition to those within scope of Phase 1B, Calypso will continue to serve as the settlement engine for these trades.

2.2 Murex and Chameleon Interface

2.2.1 Solution Design

To enable the update of TS accounts in Chameleon, MX will need to interface with Chameleon and leverage payment events to generate funding request.

Below is a schematic illustrating the process flow between Murex and Chameleon for external trades. An external trade is defined as a scenario where RMB enters into a transaction with an external counterparty and settles in ZAR or CCY. Each of these settlements, against the external counterparty, is funded via the settlement process.



Process Description:

- Traders capture trades in Murex
- The trade is matched in Murex once counterparty confirmation is received. Cash payment messages are then released for swift instructions on value date.
- Upon acknowledgement of the swift message, the payment is set to a "ToBeSettled" status and Murex sends a cash payment message to the ESB (Enterprise Service Bus).

- If the Closing Entity on the cash payment message is not London (CE <> “FRB London”), the ESB transforms the message into a Chameleon friendly format and generates a funding request. If the Closing Entity on the message is set to London (CE = “FRB London”) then the process described in Section 2.5 is triggered.
- The ESB uses PHS to determine the Chameleon portfolio name based on the portfolio name included on the incoming Murex cash payment message.
- The funding service in Chameleon receives the request, updates the TS account on a portfolio to reflect the effect of the funding request and then generates front-end and back-end funding business events for RADA. Each of these funding events has a unique funding reference, which is used as the cross-reference for recon matching in downstream reporting.
- Funding references generated of a funding request are then sent to the ESB for processing into Murex.
- Murex will then generate EOD cashflow and payment business events for RADA that will include all cross-references. The payment event must contain the nostro account name to facilitate posting to the correct GL nostro account.

For Prime Broking clients a settlement instruction will be generated as opposed to a swift message as the client accounts are housed in Chameleon. The settlement instruction will follow the process above to interface with Chameleon and generate a funding request.

Murex may net settle transactions against the same counterparty, which contain the same instrument and currency. For downstream reporting Murex will undo the netting process and generate individual cashflow and payment events that are reconcilable to individual funding events, via the unique funding reference. In this scenario, Chameleon will still generate funding events at a gross level that are reconcilable to individual cashflow and payment events

With the exception of the Group Treasury funding desk and the RMB FX desk, no GM department has a mandate to hold foreign currency exposure. All currency positions in Murex will have an internal trade with the RMB FX desk to fund their currency position. Consequently, all TS currency balances are kept close to zero in Chameleon.

Required changes:

- Murex
  - Develop a process to send a cash payment message to the ESB in the format outlined in APPENDIX A.
  - Nostro account names to be setup in Murex
  - Configuration required to receive funding reference from Chameleon as depicted in APPENDIX B
  - Generate cashflow and payment business event files for RADA that include funding references and conform to the RADA MDR.
- ESB
  - Transform cash payment message from Murex to a Chameleon funding request
  - Write back funding reference from Chameleon to Murex
- Chameleon
  - Funding service will need to be updated to receive funding requests from Murex

Open Items:

- *Is the EX number still required by Chameleon on the funding request? If it is needed we will need to include a mapping table in Murex that maps a nostro account name to an EX number*
- *Murex has a limitation against the number of characters defined in a portfolio name. In order to align the naming convention between Murex and Chameleon, all Murex portfolios will have to be configured in PHS and the mapping between the two systems will be used in the ESB.*

2.3 Internal Trades

Internal trades are defined as scenarios when RMB or Group Treasury internal departments transact with each other and settle in ZAR and CCY. In order to correctly update the respective TS account on each portfolio, every internal settlement will need to funded via Chameleon.

The funding of internal trades will follow the solution defined in Section 2.2 of this document. However, Murex does not treat external trades the same way as internals as there is no trade matching or swift acknowledgement on an internal. Murex will therefore require additional technically configuration to support the proposed funding solution.

Required changes:

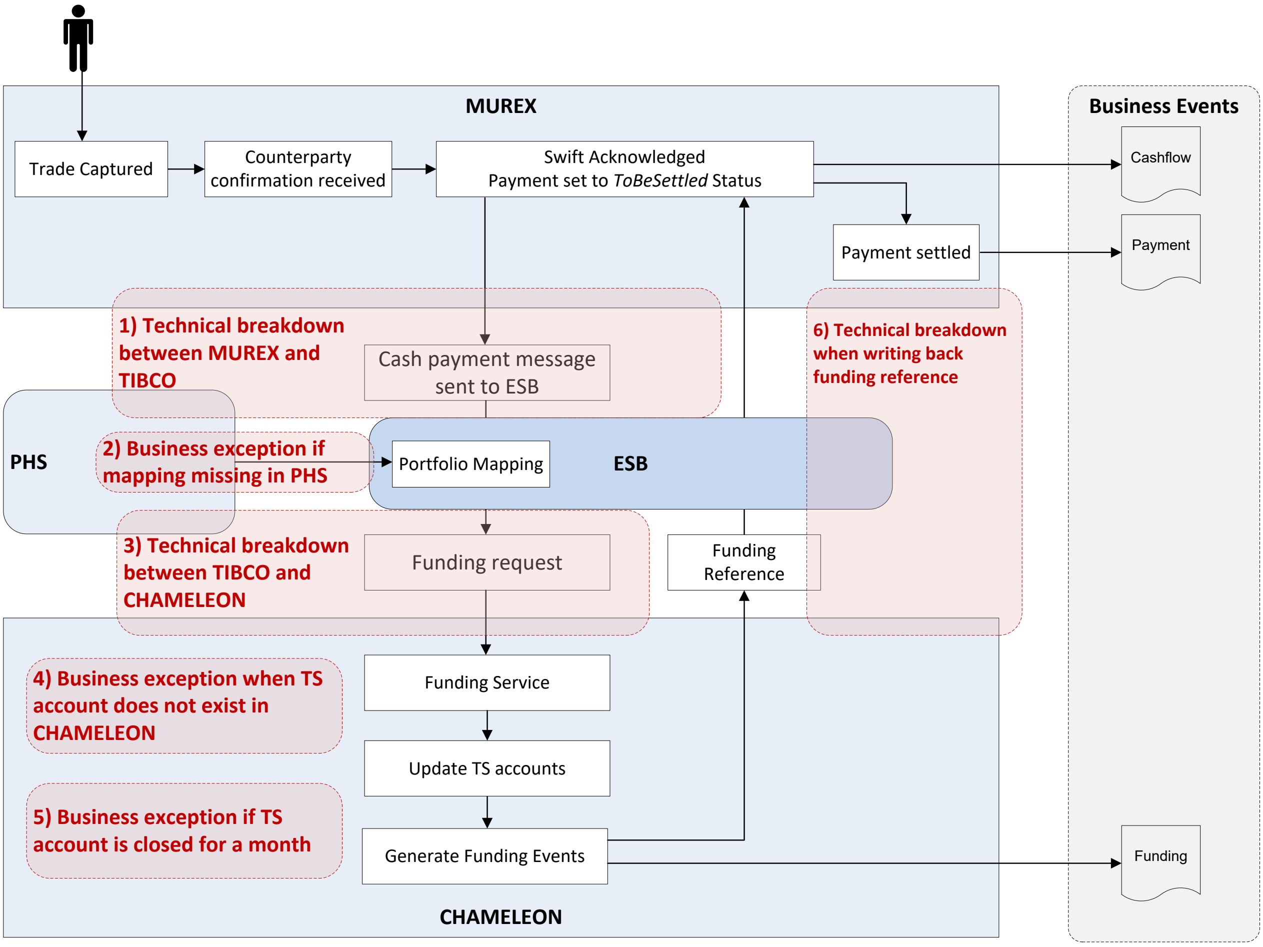
- Murex: Simulate technical workflow for swift acknowledgement

Open Items:

- *Murex to confirm if it will be possible to unwind the interim internals solution, defined here, at a later stage as it may not be required once the target funding solution is on Murex.*

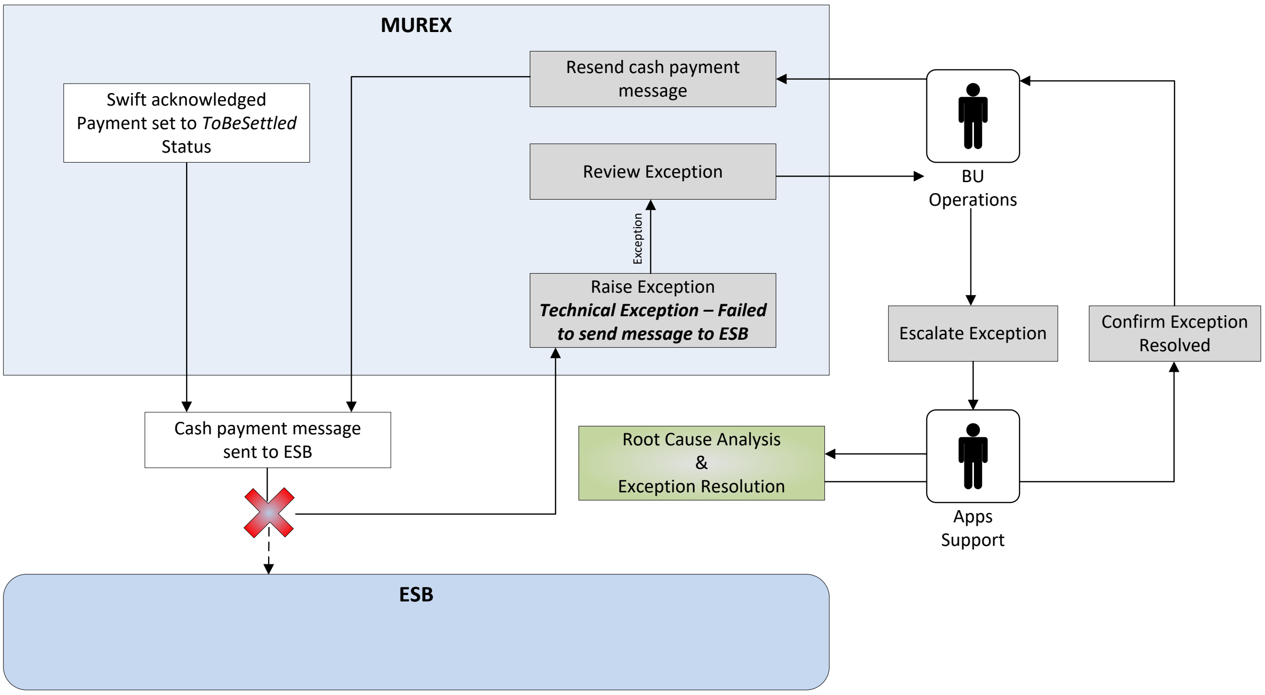
2.4 Exception Management

Below is a schematic outlining the areas of technical and business breakdown in the solution:


MUREX
Trade Captured
Counterparty confirmation received
Swift Acknowledged Payment set to *ToBeSettled* Status
Payment settled
Business Events
Cashflow
Payment
1) Technical breakdown between MUREX and TIBCO
Cash payment message sent to ESB
6) Technical breakdown when writing back funding reference
PHS
2) Business exception if mapping missing in PHS
Portfolio Mapping
ESB
3) Technical breakdown between TIBCO and CHAMELEON
Funding request
Funding Reference
4) Business exception when TS account does not exist in CHAMELEON
Funding Service
Update TS accounts
5) Business exception if TS account is closed for a month
Generate Funding Events
Funding
CHAMELEON

2.4.1 Murex to ESB

There may be instances where cash payment messages fail to load into the ESB. An exception management process will be needed in in Murex to report these failures. The process below defines the process for this scenario and must be reviewed throughout the day.

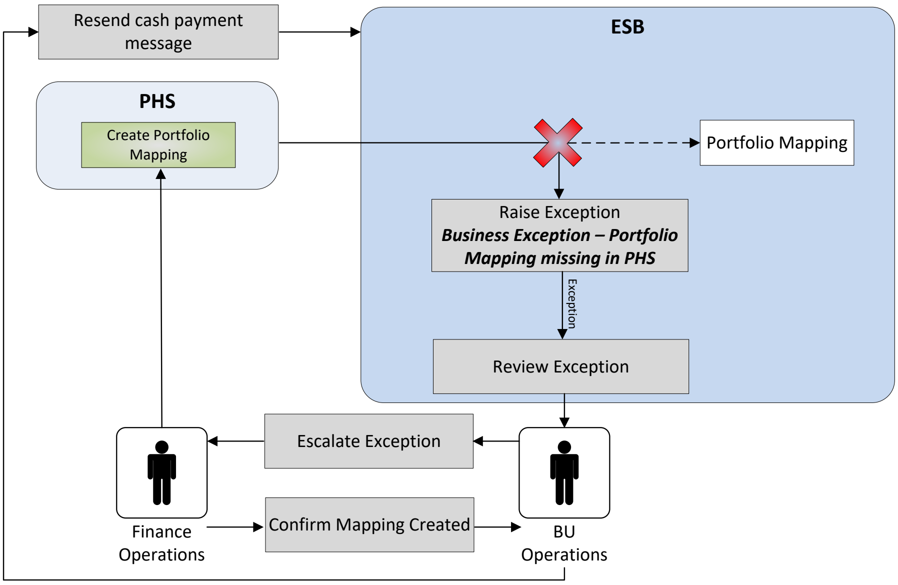

MUREX
Swift acknowledged Payment set to *ToBeSettled* Status
Resend cash payment message
Review Exception
Exception
Raise Exception ***Technical Exception – Failed to send message to ESB***
BU Operations
Escalate Exception
Confirm Exception Resolved
Cash payment message sent to ESB
Root Cause Analysis & Exception Resolution
Apps Support
ESB

Process description:

- Exception raised if Murex is unable to connect to ESB and send through the cash payment message.
- BU Ops reviews the exception in Murex and escalates exception to application support for resolution
- App support performs root cause analysis on the exception, resolves the underlying issue and then sends confirmation to BU Ops.
- BU Ops subsequently resends the cash payment message to the ESB

2.4.2 PHS Missing Portfolio Mapping

Should a portfolio mapping for Murex and Chameleon not exist in PHS then the ESB will be unable to process the funding request into Chameleon. The exception management process to manage missing portfolio mappings is described below and will be reviewed throughout the day.

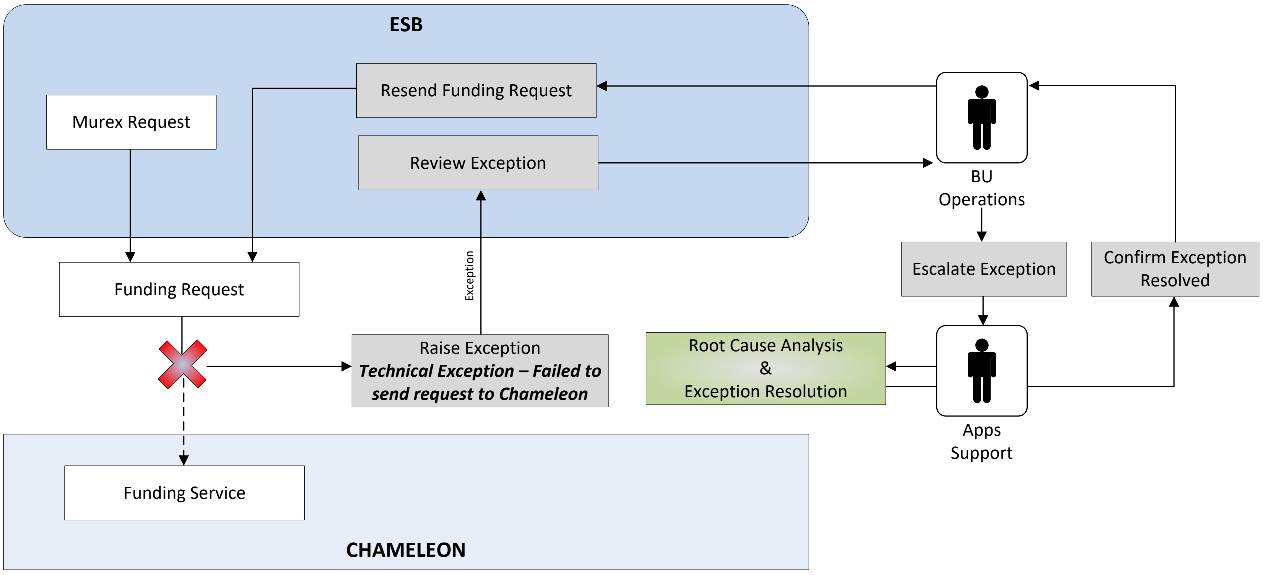


Process description:

- Exception raised if portfolio mapping does not exist in PHS
- BU Ops reviews the exception in the ESB and sends a request to Finance Operations to create the portfolio mapping in PHS
- Finance Operations creates the portfolio mapping in PHS and sends confirmation to BU Ops
- BU Ops thereafter reprocesses the cash payment message in the ESB

2.4.3 ESB to Chameleon

Exceptions may arise between the ESB and Chameleon interface due to technical breakdowns. The process below defines the exception management process for this scenario and will be reviewed throughout the day.



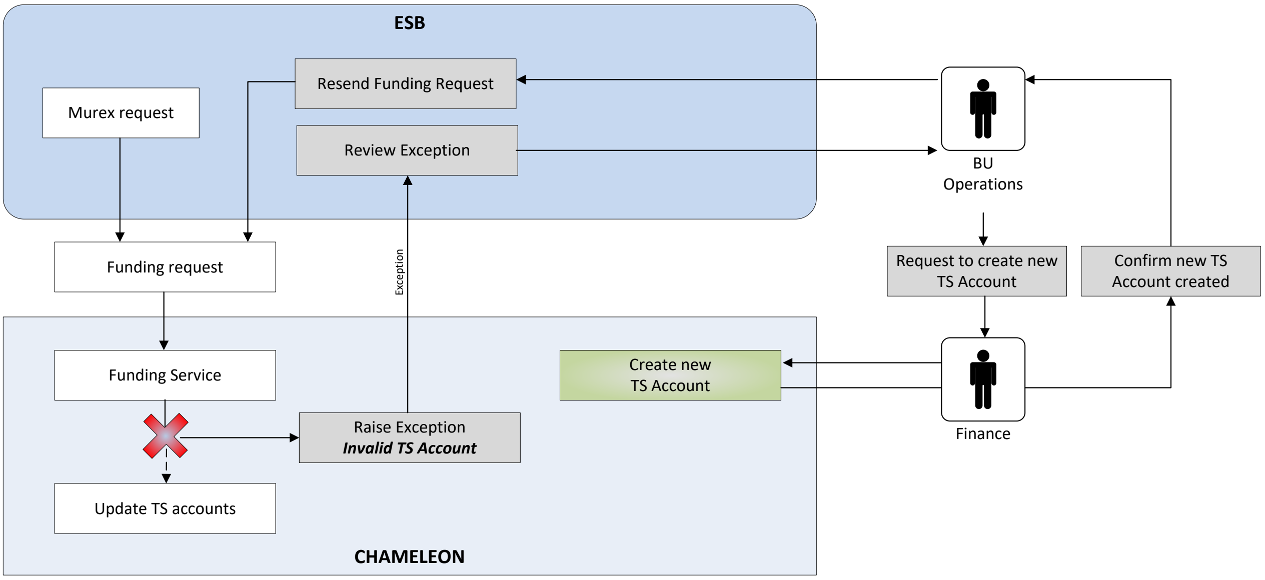
Process description:

- Exception raised if ESB is unable to send funding request to Chameleon
- BU Ops reviews the exception in ESB and escalates exception to application support for resolution

- App support performs root cause analysis on the exception, resolves the underlying issue and then sends confirmation to BU Ops.
- BU Ops then resends the funding request to Chameleon

2.4.4 Invalid TS Account in Chameleon

Instances may arise when a cash payment message is sent from Murex to Chameleon on a portfolio that does have a valid TS account in Chameleon. The process below defines the exception management process for this scenario and will be reviewed throughout the day.

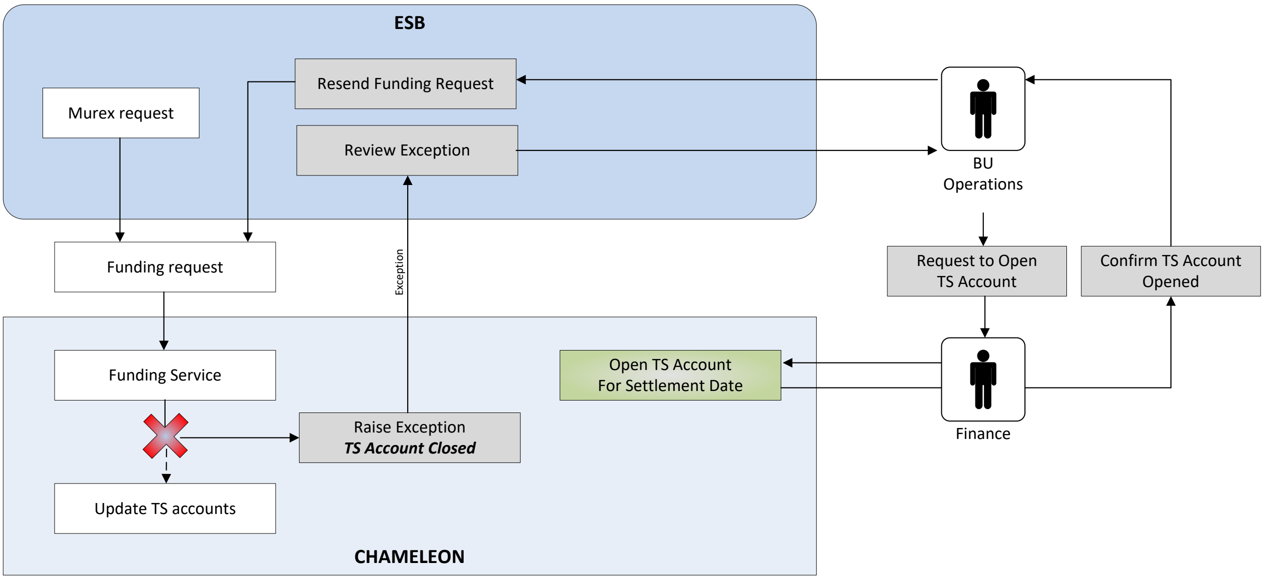


Process description:

- Exception raised if TS account does not exist for portfolio included in the funding request
- BU Ops reviews the exception in ESB and sends a request to the finance team to setup the TS account in Chameleon
- Finance configures the new TS account in Chameleon and sends confirmation to BU Ops
- BU Ops thereafter resends the funding request to Chameleon to impact the new created TS account

2.4.5 TS Account Closed in Chameleon

Chameleon closes all TS accounts at the end of a month. To facilitate back dated settlements, the last business day of the previous month is kept open for the first three business days of the new month. Any settlements that occur in the first three business days of the new month may be backdated to the previous month. An exception will be raised in the event that a funding request is received for a back date for which the TS account is closed in Chameleon. The process below defines the exception management process for this scenario and will be reviewed throughout the day.

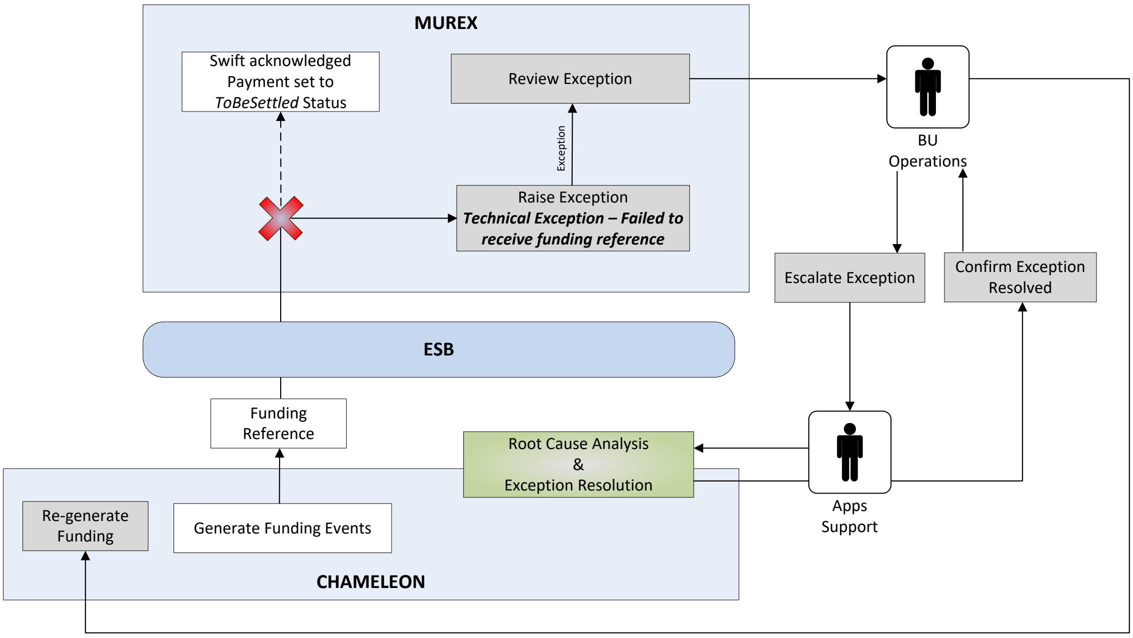


Process description:

- Exception raised if TS account is closed for a date included in the funding request
- BU Ops reviews the exception in the ESB and sends a request to the finance team to open the TS account in Chameleon for date on the funding request
- Finance opens the TS account in Chameleon and sends confirmation to BU Ops
- BU Ops thereafter resends the funding request to Chameleon to impact the date reflected on the funding request

2.4.6 Funding Reference write-back Exception

An error may occur when writing back the funding reference to Murex and the funding reference transfer attribute will show the error received. The process below defines the exception management process for this scenario and will be run throughout the day.


MUREX
Swift acknowledged Payment set to ToBeSettled Status
Review Exception
BU Operations
Exception
Raise Exception
Technical Exception – Failed to receive funding reference
Escalate Exception
Confirm Exception Resolved
ESB
Funding Reference
Root Cause Analysis & Exception Resolution
Apps Support
Re-generate Funding
Generate Funding Events
CHAMELEON

Process description:

- Exception raised if error on funding reference received in Murex
- BU Ops reviews the exception in Murex and escalates exception to application support for resolution
- App support performs root cause analysis on the exception, resolves the underlying issue and then sends confirmation to BU Ops.
- BU Ops then regenerates the funding reference in Chameleon

2.5 London

As part of Phase 1B all London Fixed Income and IRD trades will be captured on Murex. Integra-T is the current trading system for all London, with the funding process being replicated within the system via call accounts. Call accounts are equivalent to TS accounts that are housed in Chameleon.

2.5.1 Call Account Mapping Table

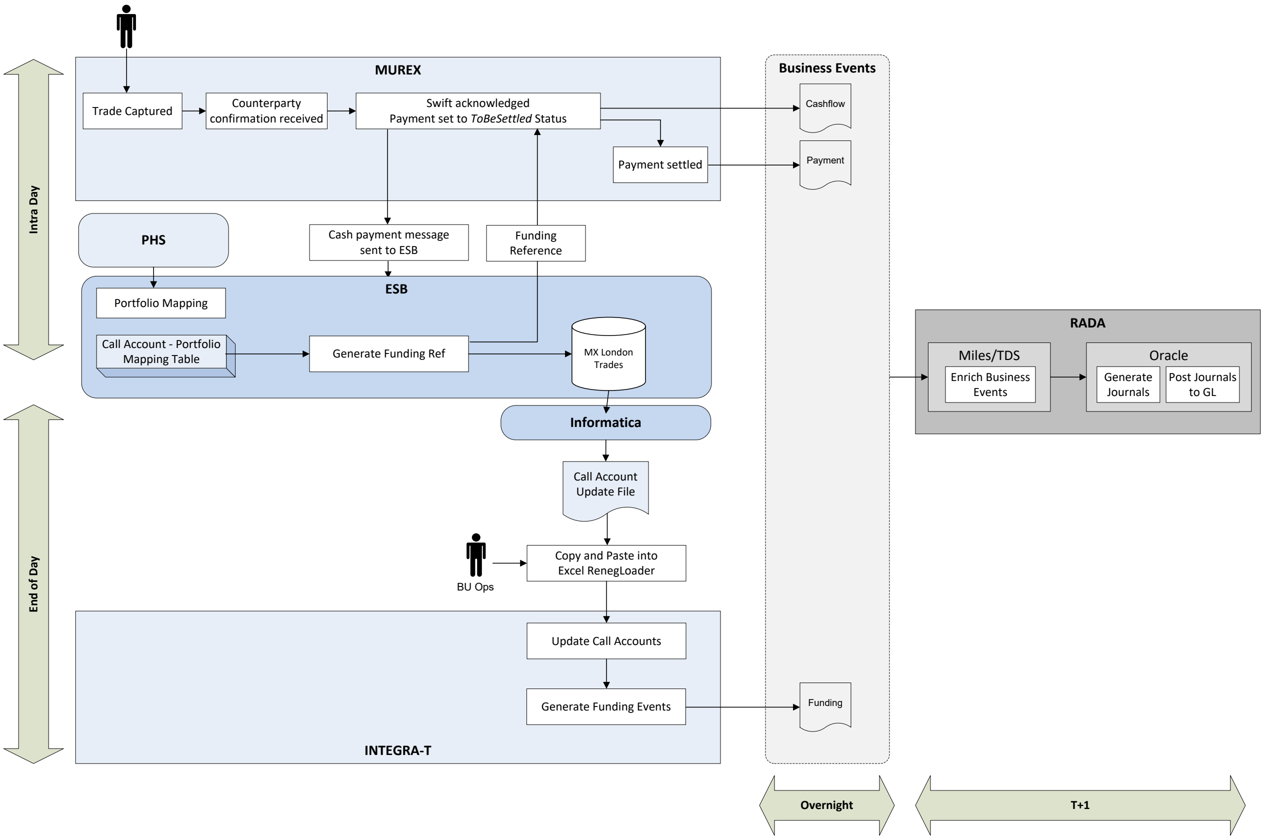
Unlike Chameleon, Integra-T is unable to determine the call account from the incoming Murex messages that includes information on the portfolio and currency. In order to update the correct call account in Integra-T, a mapping is required between portfolio and currency pairs to call accounts. The mapped call account will be included in a CSV file that will be uploaded into Integra-T. Any new call accounts created in Integra_T will have to added to the call account mapping table in the ESB. The update of new mappings will be managed by BU Operations.

Below is the structure of the call account mapping table:

Portfolio	Currency	Call_Account_No
FDCM-FIXIN	USD	101493

2.5.2 Solution Design

As part of the Phase 1B implementation, the update to call accounts in Integra-T will be facilitated via the process below:



Process Description:

- Traders capture trades in Murex.
- The trade is matched in Murex once counterparty confirmation is received. Cash payment messages are then released for swift instructions on value date.
- Upon acknowledgement of the swift message, the payment is set to a “ToBeSettled” status and Murex sends a cash payment message to the ESB.
- If the Closing Entity on the cash payment message is set to London (CE = “FRB London”), then the ESB generates a funding reference and enriches the cash payment message with a call account number derived of the mapping table stored in the ESB.
- The funding reference generated in the ESB is then sent to Murex, for inclusion in the RADA cashflow and payment business event files.
- Events generated of the cash payment messages are stored in a database in the ESB using PHS to determine the Integra-T portfolio name based on the portfolio name from the incoming Murex cash payment message.
- At 17:00 (SA Time), all events written to the database for a day are processed via Informatica into a CSV file. This CSV file will contain attributes received in the cash payment message, as well the mapped call account and the ESB generated funding reference.
- The CSV filed will be saved in the following directory: IntegraT(\\intium) (R:) /Reneg Loader/Murex
- BU Ops will use events in the CSV file to update the Reneg Loader, which subsequently updates the call accounts in Integra-T.
- Funding business events for London trades will be generated out of Integra-T as part of the EOD process.
- Murex will generate cashflow and payment business events for RADA that will include the funding reference generated in the ESB.

Required changes:

- ESB
  - Create and manage a one-to-one mapping table between portfolio and currency to call accounts as defined in Section 2.5.1
  - Generate unique funding reference of Murex cash payment message
  - Store London events in a database. The database attributes can be found in APPENDIX C
- Informatica
  - Map events from database into CSV file as defined below
- Integra-T
  - Absorb the funding reference from the comment field in the CSV file
  - Update the cashflow/payment to funding recons to exclude Murex trades as Murex cashflow and payment events will not be in Integra-T

2.5.3 Integra-T CSV Input File

As part of the EOD of day process, Informatica will provide BU Ops with a CSV file, containing all London Murex trades, which will be processed into the Reneg Loader.

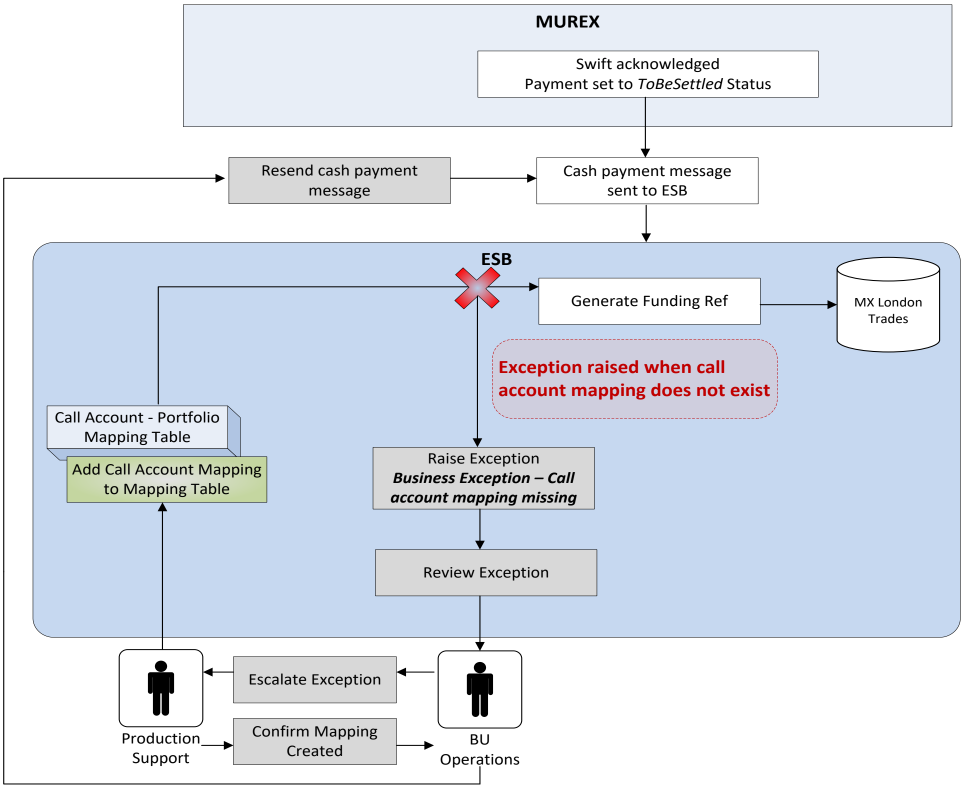
Below is a mapping between the attributes in the ESB database and the CSV file output:

ESB Database	Relationship	CSV File
reference	None	
sourceSystem	None	
portfolioCode	Mapped To	BUnit
valueDate	Mapped To	ValueDate
currencyCode	Mapped To	Curr
amount	Mapped To	Amount
payOrReceive	Mapped To	Direction
nostro	None	
originatingReference	Mapped To	Comments
originatingSourceSystem	None	
paymentReference	None	
cashflowType	None	
Closing_entity	None	
FundingReference	Mapped To	Comments
Call_Account_No	Mapped To	CallAccNo
Counterparty	Mapped To	Comments

** The "comments" field in the CSV file will be concatenated as follows: originatingReference /Counterparty/FundingReference*

2.5.4 Exception Management

There may be instances where a portfolio and currency pair on a cash payment message does not have a mapping in the mapping table in the ESB. An exception management process will be needed to cater for these exceptions and will be managed by BU Ops as part of the EOD process. Below is a schematic detailing the exception management process:

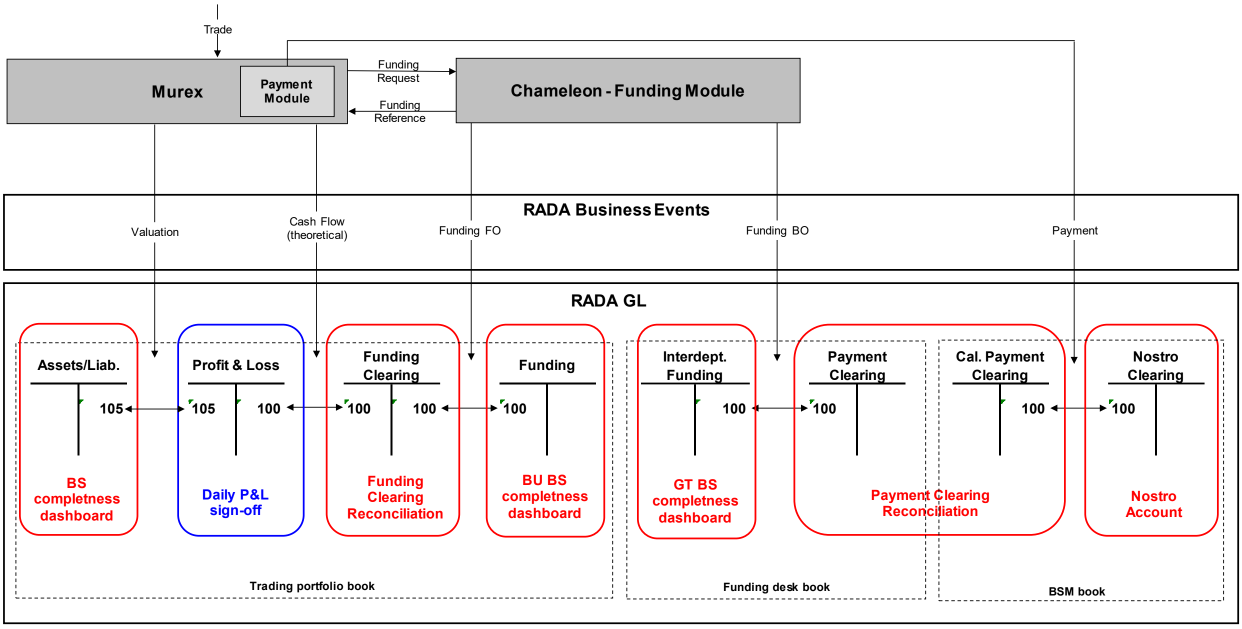

MUREX
Swift acknowledged
Payment set to *ToBeSettled* Status
Resend cash payment message
Cash payment message sent to ESB
ESB
Generate Funding Ref
MX London Trades
Exception raised when call account mapping does not exist
Call Account - Portfolio Mapping Table
Add Call Account Mapping to Mapping Table
Raise Exception
Business Exception – Call account mapping missing
Review Exception
Escalate Exception
Confirm Mapping Created
Production Support
BU Operations

Process description:

- Exception raised if a call account mapping does not exist for a portfolio and currency pair
- BU Ops reviews the exception in the ESB and sends a request to the Production Support to add the new call account mapping to the mapping table
- Production Support adds the new call account mapping and sends confirmation to BU Ops
- BU Ops subsequently resends the cash payment message to the ESB

2.6 RADA Integration

RADA serves as the banks financial control and reporting process. A key aspect in RADA is ensuring the completeness, validity and accuracy of the financial data in GL. The following schema outlines how different business events are accounted for in the Oracle GL and provides an overview of daily controls that are in place in the RADA solution to validate the resulting GL balances.



RADA Business Events

Murex will provide a set of business events to RADA (transaction, valuation, cashflow, payment) according to the MDR daily by 22:00 (Minimum Data Requirements). These events will be batched processed into RADA as part of the overnight run. RADA will enrich and standardise these business events and account for them in the bank's GL.

As depicted in the schematic above, funding events will still be generated of Chameleon as funding requests are received from Murex. Once a funding request is acknowledged, the corresponding funding reference will be sent to Murex to be included in the cashflow and payment business event files for recon matching.

The following controls are embedded in the RADA architecture to ensure the completeness, validity and accuracy of data in the GL:

- **Funding clearing reconciliation**: this reconciliation ensures that each cashflow must have a corresponding funding event to cater for its funding implications. This is a key control in the GL and is validated for each portfolio and should square to a zero balance every day. To facilitate exception resolution for when these accounts do not square to zero, the RADA solution uses reconciliations of cashflows / fundings in the bank's reconciliation application - Intellimatch. IntelliMatch uses the cross references provided in business events to match cashflows to corresponding fundings and escalates any unmatched exceptions to the relevant Ops team to resolve.
- **Payment clearing reconciliation** – this reconciliation ensures that for all cash-flows that emanate in source system a corresponding payment has been processed. This reconciliation is also performed daily in Intellimatch and any exceptions are managed by the central Reconciliation team.
- Several other dashboards exist in the OBIEE reporting application within RADA to ensure completeness of data between business events and the GL.

Required changes:

- Murex: Process required to generate transaction, valuation, cashflow and payment business event files that conform to the RADA MDR. This must include all cross references received from Chameleon to be leveraged in IntelliMatch.
- RADA
  - Update PHS and TDS mappings to receive business events from Murex.
  - Update Control-M process Murex business event files
- IntelliMatch: Update recon extracts to include Murex cashflow and payment events.

Backdating principle in RADA:

- To provide necessary daily financial control, once the daily P&L for a particular day has been approved in Oracle GL, the day is closed for posting.
- Any cashflow, funding, journal, payment events backdated to the already closed day, will be posted in Oracle GL with the posting date equal to the next open business day in Oracle GL.
- On a month end, the last day of the month remains open for the posting for the first 4 days of the month to enable to post backdates as part of the month end close process.

APPENDIX A: FUNDING REQUEST MDR

Attribute	Description	Format	Sample Data
reference	A unique reference per request from the requesting system	string	164449420
sourceSystem	Requesting system identifier	string	CAL
portfolioCode	The portfolio code that the request is for	integer	194687
valueDate	The date that the funding should happen	date	20/01/2017
currencyCode	The currency code that this request is for	string	GPB
amount	The amount that this request is for	decimal	9051.79
payOrReceive	The direction of the funding request	string	Pay
nostro	The nostro account code that the request should settle over	string	EX10E00081
originatingReference	The reference of the originating system	string	184947953
originatingSourceSystem	The identifier of the originating system	string	CAL
paymentReference	The payment system reference	string	Other
cashflowType	The type of cashflow of the funding request	string	RB17BV1583
Closing_entity	The closing entity that the portfolio belongs to.	string	FRB London
Counterparty	The counterparty name against which the trade is booked	string	DE BEERS

APPENDIX B: FUNDING ACKNOWLEDGEMENT

Attribute	Description	Format	Sample Data
sourceSystem	Requesting system identifier	string	CAL
sourceSystemTransactionReference	A unique reference per request from the requesting system	string	810570561
status	The date that the funding should happen	string	SUCCESS
message	The currency code that this request is for	string	2017-02-20T13:15:12.053+02:00
fundingReference	A unique reference assigned to event in Chameleon	string	48302980

APPENDIX C: ESB DATABASE STRUCTURE FOR LONDON TRADES

Attribute	Description
reference	A unique reference per request from the requesting system
sourceSystem	Requesting system identifier
portfolioCode	The portfolio code that the request is for
valueDate	The date that the funding should happen
currencyCode	The currency code that this request is for
amount	The amount that this request is for
payOrReceive	The direction of the funding request
nostro	The nostro account code that the request should settle over
originatingReference	The reference of the originating system
originatingSourceSystem	The identifier of the originating system
paymentReference	The payment system reference
cashflowType	The type of cashflow of the funding request
Closing_entity	The closing entity that the portfolio belongs to
FundingReference	The ESB generated funding reference
Call_Account_No	The mapped call account derived of the mapping table in the ESB
Counterparty	The counterparty name against which the trade is booked