# Reference documentation

|  |  |  |
| --- | --- | --- |
| Reference | Document | Description |
| **Jira** | https://helpdesk.clear2pay.com/helpdesk/browse/COBAXTRM-71 | The interaction between the bank and C2P regarding the logic of the start/end of day trigger. |
| **Parsing service AD** | COBAXT - AD - Parsing Service.docx | Pain 001 mapping approach |
| **User case business reference** | COBA\_PARSER\_BS\_UC001 - Interchange Parser Service.doc | The logic for the generation of business reference. |
|  |  |  |
|  |  |  |

Table 1: Reference documentation

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| **1.0** | **29-01-2015** | **Michael Couck** | Initial draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 1: Revision History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Reviewer** |
|  |  |  |
|  |  |  |

Table 1: Review History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Date | Reporter | Description | **Resolution** | **Status** |
| **OP1** |  |  |  |  | OPEN |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table 1: List of Open Points

# Overview

The business reference that the bank wold like, is generated, using a sequence, per day, and the date. This data must be available for the reporting. The structure of the business reference is referred to in the referenced documents, listed above.

## Purpose

This document is to finalize the logic and functionality required by the bank as the logic and timing has some points of interest, one of which is the date that is used to generate the business reference.

## Scope

An MDB is triggered by UC4 on the bank side, COBAXTDailySequencerRollerMDBean. This MDB updates the date in the database that is used for generating the business reference. The AssignBusinessReference needs to be updated, to use the date persisted in the database rather than use the business calendar for the date to be used in the generation of the business reference. A message is put on the de-warehousing queue to trigger the de-warehousing.

## Existing Situation

Currently the date used in the generation of the business reference is the one from the calendar. The new logic will avoid that there are duplicate business references, and that the de-warehousing can be called multiple times per day without affecting the date that is used for the business reference.

## Target Situation

The date used for generating the business reference comes from the database, and we reset this with a JMS trigger. The date and the sequence can be reset once per day, but the de-warehousing service can be called several times a day.

# Business View

This change in the logic, to put the date in the database for the business reference, will ensure that the business reference is always unique, and confirms to the request of the bank.

## Functional Description

A date is put in the database, and updated when the MDB is triggered. If the date in the database is not equal to the next business day then the next business day is used to update the date in the database, and the sequence is reset to 1. If the date in the database is already the next business day then there is no date update and no sequence reset.

The de-warehousing is always called regardless of the date.

The timing logic is duplicated from the Jira here as requested.

Day zero is 02-01-2015, next business day(n + 1) is 05-01-2015:  
1) Before cut off & before trigger(default)  
Result:  
Processing dates will be 02-01-2015  
Business reference date 02-01-2015

Important, means that we DO NOT change the business reference for instructions that are warehoused between the cut off and the trigger  
2) After cut off & before trigger  
Result:  
Processing dates will be 02-01-2015  
Business reference date 05-01-2015

3) After cut off & after trigger & before 12:00  
Result:  
Processing dates will be 02-01-2015  
Business reference date 05-01-2015

4) Trigger at 3:00 pm cut off at 6:00 pm (edge case)  
Result:  
Processing dates will be 05-01-2015  
Business reference date 02-01-2015

## Business Constraints and considerations

N/A

## Global Functions Description

\* COBAXTDailySequencerRollerMDBean – persists the business reference date in the database

\* AssignBusinessReference – creates a business reference for transactions, using the date and a sequence, conforming to a particular format

\* De-warehousing queue – accesses the message deliverer to post a message on the queue

## Functionality Components Mapping

N/A

# Software Model

## Non Functional Requirements and Technical Constraints

|  |  |
| --- | --- |
| **Requirement ID** | **Description** |
| **NF\_REQ\_01** | As the reports will be posted on a queue, the size of the report is important. Consequently if there are too many instructions in one report, the report must be broken into several reports to facilitate posting on the queue, i.e. less than 1 meg report sizes. |

Table 1: Non-Functional Requirements

## Components

### List of components on server side

Please refer to section 3.3 for the components that are involved in the solution and a brief description.

### List of components on client side

N/A

## Software Architecture

### MDB (COBAXTDailySequencerRollerMDBean):

#### Description

This MDB is responsible for updating the business reference date in the database, and resetting the sequence that is used to generate the business reference.

#### Processing Logic

The next business calendar date is accessed using the calendar and checked against the business reference date in the database. If these are different then update the date in the database with the next business date, reset the sequence to 1 and post a message on the de-warehousing queue to trigger the dewarehousing. If the next business date and the date in the database are the same then just post a message on the de-warehousing queue.

#### (Optional) Message layout

N/A

#### (Optional) Parameters

The message for the de-warehousing queue must contain parameter 'releaseAll', set to 'instructions & transactions'. This will trigger both instructions and transactions to be de-warehoused.

#### (Optional) Discussion

The transaction boundaries for the incoming daily sequence roller will encompass the transaction for de-warehousing. These need not be in the same transaction, but does have impact if either of these fail, i.e. the database update or posting to the de-warehousing queue. The database could continuously fail, meaning that the de-warehousing would never be called.

## Software Organisation

|  |  |
| --- | --- |
| **Project Name** | **Description** |
| **COBAXT** | The project base logic |

Table 1: project locations

## Impacts

### Data access layer

The data store table will be used, the date stored as a string value. Objects to access the database are the InterfacingDataAccessObject and the Criteria for querying the table.

Classes: com.clear2pay.bph.opfpayment.dataaccess.dao.InterfacingDataAccessObject, com.clear2pay.bph.opf.search.Criteria

Parameters for query: business-reference(for key in COBADATASTORE)

Table value: String date(format 'yyyy-MM-dd', as this seems to be most common format in the OPF layer)

#### Scripts

N/A

#### Clean-up

N/A

### EJBs / MDBs

No new MDBs or EJBs.

### BPEL flows

No impact.

### Testing (QA)

#### QA

QA should verify that the business day is updated to the following business day, and that the sequence is updated, in the case the business day is different from the calendar business day. And in the case it is not, then there is no change in the business day in the database and the sequence is not reset.

There should always be a message on the de-warehousing queue regardless of the date being updated in the database, and seperate from the update of the sequence.

As always larger volumes need to be tested, i.e. more than 100 000 transactions to be reported on, and more than 100 000 transactions to be de-warehoused.

#### NFT

As for QA, larger volumes need to be tested for performance, more than 100 000 transactions to be reported on and 100 000 transactions to be de-warehoused. This should result in load on the queues, specifically the transaction report queue, which is the focus of this logic.

### Application Server configuration

No extra configuration, the queue for the transaction report is already defined as XMLReportQ. This queue must be re-defined as TransactionReportQ to be in synch with the change in the name of the report.

### SCM

Scripts that need to be updated:

\* configure\_wps.jacl – Update XMLReportQ to TransactionReportQ

\* accounting-cobaxt.xml - Update XMLReportQ to TransactionReportQ

\* servicelocator-cobaxt.xml - Update XMLReportQ to TransactionReportQ

\* ibm-ejb-jar-bnd.xmi - Update XMLReportQ to TransactionReportQ

\* cobaxt\_was\_queue\_list.py - Update XMLReportQ to TransactionReportQ

\* ejb-jar.xml - Update XMLReportQ to TransactionReportQ