CS\_RTRN\_BS\_UC004 – Invalidate a Transaction

# Brief Description

This Use Case depends on “**SEPA\_RTRN\_BS\_UC004 – Invalidate a Transaction**”, and enables an actor to invalidate an indicated Transaction.

This use case only details the differences that are required to the **SEPA** use case. All functionality documented in the **SEPA** use case is assumed to be available in this use case.

# Actors

## Primary Actors

No Change

## Secondary Actors

# Conditions

## Pre-Condition

No Change

## Post Conditions on success

Added condition 5

1. An informational worklist item is created to record that the Transaction has been successfully invalidated.

## Post Conditions on Failure

# Trigger

No change

# Main Flow: CS\_RTRN\_BS\_UC004.MF – Invalidate a Transaction

Following steps have been added.

1. If transaction is an Outbound Direct Debit Transaction, then the **system** updates the CreditLimit table with the details of Credit Party, Settlement Amount and Settlement Day for the Transaction

IF PaymentTransaction.outbound = true AND PaymentTransaction.incomingInstruction.instructionType = “pain.008\*”

THEN

CreditLimit.creditPartyId = PaymentTransaction. creditPartySchemePartyId

CreditLimit.processingScheme = PaymentTransaction.processingScheme

CreditLimit.settlementDay = PaymentTransaction.settlementDay  
(OR PaymentTransaction.incomingSettlementDay if PaymentTransaction.settlementDay is empty)

CreditLimit.settlementAmount = - PaymentTransaction.settlementAmount (negative settlementAmount)

CreditLimit.processingAgreement = (original)PaymentInstruction.usedAgreement

Note: if one of the fields is not filled, no record will be created.

1. The system creates a worklist item - CS\_WLI\_RTRN\_BS\_UC001\_01 to record that the Transaction has been successfully invalidated.
2. The **system** calls *CS\_ACCNTG\_BS\_UC001 - Send Accounting Events* with the following two parameters:

* The payment entity;
* Id For Market Side MS9001;
* Id For Client Side If PaymentTransaction.outbound == TRUE AND PaymentTransaction.classification == “CreditTransfer” then CS9003 else CS9001.

# Alternative Flows

## Alternative Flow 1: CS\_RTRN\_BS\_UC004.AF01 – Cancel is invalidated after reconciliation with original Transaction.

No change

## Alternative Flow 2: CS\_RTRN\_BS\_UC004.AF02 – Cancel mandate version.

No change

# Extension Points

## Extension Point 1: CS\_RTRN\_BS\_UC004.EP01 – Enrich original transaction with R-Transaction attributes

No change

## Extension Point 2: CS\_RTRN\_BS\_UC004.EP02 – Create Accounting Event

No change

# Business Rules

## CS\_RTRN\_BS\_UC004.BR01 – Computation of periodSequence

No change

# Notes / Assumptions

# System Messages

No change

# Issues

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No: | Description: | Date | Action: | Status |
|  |  |  |  |  |

# Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Rev | Who | Reference | Description (Table Heading) |
| 16/12/2013 | 0.1 | BCA | CSIDEV-92 | Created |
| 09/05/2014 | 0.2 | BCA | CSIDEV-672 | Added condition for the accounting event |
| 20/06/2014 | 0.3 | AVI | CSIDEV-1272 | Added step 65 in MF |
| 03/07/2014 | 0.4 | BVC | CSIDEV-1473 | Removed ‘-‘ from accounts. |
| 26/08/2014 | 0.5 | BVC | CSIDEV-1899 | Accounting events aligned with ones in IP.  (added SDD and changes SCT) |
| 07/10/2014 | 0.6 | AVI | CSIDEV-2505 | Updated Step 65 in MF |
| 16/10/2014 | 0.7 | RZW | CSSICI-573 | Refactoring of the accounting design. Centralising all accounting logic in the accounting service. No functional changes. |
| 27/11/2014 | 0.8 | RZW | CSSICI-764 | Moved AE from issue processor to here. |
| 25/02/2015 | 0.9 | BVC | CSIDEV-4044 | Use incomingsettlementday if Settlementday is empty. |

# Reference Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| Document Name | Version | Description | Location |
|  |  |  |  |