PTS/SMV
Cloud Service Manual

Version 10.0.x

Contents

1 IP Protection

Trademarks

CeleritiFinTech, the *CeleritiFinTech logo* and *celeritiFT.com* are trademarks or registered trademarks of DXC Technology. Other product and service names might be trademarks of CeleritiFinTech or other companies.

Oracle and *Java* are registered trademarks of *Oracle* and / or its affiliates.

Apache Wicket, *Wicket*, *Apache log4j*, *Apache*, the *Apache feather logo*, and the *Apache Wicket project logo* are trademarks of *The Apache Software Foundation*.

iText® is a registered trademark of *iText Group NV* (formerly known as 1T3XT BVBA), the company responsible for ongoing development and commercial support of the *iText Core Library*.

Jaspersoft, the *Jaspersoft logo*, *Jaspersoft iReport Designer*, *JasperReports Library*, *JasperReports Server*, *Jaspersoft OLAP*, and *Jaspersoft ETL* are trademarks and / or registered trademarks of *Jaspersoft Corporation* in the United States and in jurisdictions throughout the world.

Open Source Licenses

PTS/SMV is delivered with Open Source Software (OSS) components potentially developed or owned by third parties. OSS components may be used according to the terms and conditions of the specific license under which the relevant OSS components are distributed, but are provided "AS IS" by such third parties and *CeleritiFinTech*.

Open Source Software Notice

Open Source Technology	URL	License Type
Apache Commons	http://commons.apache.org	Apache License, Version 2.0
Bouncy Castle Crypto APIs	https://www.bouncycastle.org	Legion of the Bouncy Castle License
Groovy	http://www.groovy-lang.org/	Apache License, Version 2.0
Hamcrest-core	http://hamcrest.org/JavaHamcrest/	BSD-3-Clause
Itext	http://itextpdf.com/	
Jasper Reports	https://community.jaspersoft.com/project/jasperreports-library	LGPL
JavaMail	http://www.oracle.com/technetwork/java/javamail/index-138643.html	CDDL 1.1, GPL 2.0
JAXB	https://javaee.github.io/jaxb-v2/	CDDL 1.1, GPL 2.0

Open Source Technology	URL	License Type
Joda-Time	http://www.joda.org/joda-time/	Apache License, Version 2.0
JUnit	https://junit.org	Eclipse Public License 1.0
Jxl	http://jxl.sourceforge.net/	GPL 2.0
Log4j	http://logging.apache.org/log4j/1.2/	Apache License, Version 2.0
POI	https://poi.apache.org/	Apache License, Version 2.0
Shiro	http://shiro.apache.org/	Apache License, Version 2.0
Slf4j	http://slf4j.org/	MIT License
Wicket	https://wicket.apache.org/	Apache License, Version 2.0

2 Introduction

Experience with SEPA implementation shows that every version step brings test effort for format and business changes. Banks and their service providers are facing the need to generate new test files in order to accommodate the massive changes in the various SEPA clearing formats.

The SEPA message validator PTS/SMV is a testing tool, which receives SEPA messages and as a simulator of an inter-PSP clearing creates complex result messages. PTS/SMV Cloud Service (hereinafter PTS/SMV) enables business tests for SEPA regular and R-transactions in the form of a cloud-enabled service.

Normally PTS/SMV expects valid messages. In case of an invalid message upload there is a validation report. The validation report can then be downloaded by the user via the Web application.

This document describes how to use PTS/SMV for SEPA testing.

3   Supported Schemes

PTS/SMV supports EBA and Bundesbank schemes with the following EPC Rulebooks
- SCT (2021 V1.0)
- SDD CORE (2021 V1.0)
- SDD B2B (2021 V1.0)

PTS/SMV also supports *DFÜ Abkommen Anlage 3 (V3.4)* and The Deutsche Bundesbank's technical specifications for the clearing and settlement of interbank SEPA credit transfers via the RPS SEPA-Clearer 2021 version 0.9 ("SCT/SCL technical specifications")*, in addition to *Bundesbank Spezifikationen Scheck v1.2 (SVV)*.

CFT will inform all customers about upcoming rulebook versions supported by PTS/SMV.

4   Service

PTS/SMV creates validation reports for SEPA transactions and generates new transactions depending on the content of each transaction. The cloud service described implements an easily accredited web based service hosted by CFT.

This means that after a simple online registration, the user is able to send clearing payments to this new CFT cloud service using an upload/import function.

PTS/SMV supports SCT and SDD rejects and returns, as well as SDD refunds, reversals, requests for cancellation, and SCT recalls for validation reports. Regular SCTs, SCT Recalls and SDDs can be imported to generate new SEPA transactions.

4.1   Sign-Up

Users with the administrator's privilege are enabled to approve up to additional users. The administrator also can delete existing users of the organisation's account.

4.2   Upload
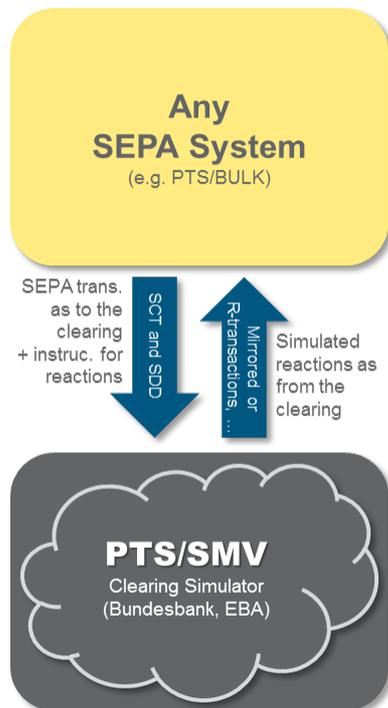
The user uploads a ZIP file containing one or more files all compliant to one SEPA version. For this, the user can select the SEPA version to be tested. The upload capacity overall is 80 MB per user.

In the cloud service dialogs, SEPA versions are only named by their general version (e.g. 3.4 is named 3) and B2B is assigned to the current CORE number (e.g. B2B 1.3 corresponds to CORE 3.4 = 3).

5   Functional Description for PACS

PTS/SMV comprises two steps - validation and simulation.

In the validation phase, payments are checked against XML rules predetermined by XSD scheme files provided by EBA and Deutsche Bundesbank. Furthermore, there are checks concerning the contents of the payments.

SEPA payments may have a production or test indicator and are free to have old value dates (or requested settlement date). So, previously used payments can be reissued to PTS/SMV and do not need to be built again by the bank's SEPA system. For this, the cloud service provides a simple online function to reissue a file to be processed. PTS/SMV generates current value dates and references to make it possible for the bank's SEPA system to receive it.

5.1   General Approach

To use PTS/SMV in tests, you need two things to be done:

I. Validation of SEPA files

1. You take the SEPA files that you want to be validated and zip them.
2. After logging on to the PTS/SMV you fill the upper form.
3. Press the *Browse Button* to select the zip file you want to import to PTS/SMV.
4. After that you select the SEPA version that is used by your files. If a file in the zip archive does not correspond to your selected version, the validation will fail.
    You can optionally type in a comment to identify your files easier later on, as this comment will be shown in the files list.
5. After committing your request, you will find the unzipped files in the files list with a status indicating when they are processed.
6. To update the files list, press the refresh-icon on the upper left of the list.

II. Generation of new SEPA files and R-Transactions

PTS/SMV is controlled by the EndToEnd IDs which can be found on transaction level in each SEPA transaction file. In tests this field is normally used to hold a test case number. The PTS/SMV functions use this field to get commands what you want to have generated – a test case number can also be added after a command. In addition, some functions are controlled by options.

Files with such prepared EndToEnd IDs have to be processed in the same way as for validation in order to produce new SEPA files ready for download.

In the following sections all functions of PTS/SMV are described in detail.

5.2   Test Functions PACS

5.2.1   Mirror (SCT, SDD, SCC, SVV)

Recipient and sender information for individual transactions are exchanged against each other before the latter are sent back to the sending institution. This simulates incoming SEPA transactions.
For this function you can use an EndToEnd ID like **ECHO**-TC00321 where TC00321 is the test case number.

5.2.2   Duplicate (SCT, SDD, SCC, SVV)

Comparable to mirroring, but original transactions are duplicated to simulate mass payment processing. This command can also be used when the number of payments within a payment file is limited. The EndToEnd ID is like Mirror.

Possible options:
Duplication Count *(*N), Transactions in File *(*F)

As an *example*, the following EndToEnd ID will generate 3 files with 30 payments of the mirrored transaction plus one file with 10 payments:
    ECHO-N:100-F:30.

5.2.3   Reject (SCT, SDD, SCC, SVV)

Provides a transaction type "reject" for the transaction sent. In addition, specific clearing reject types can be simulated by PTS/SMV (CVF, DVF or DNF). Based on the business use case, PTS/SMV will automatically identify and generate file rejects or single rejects.

For SCT:
SCT-Rejects are always resulting in CVF-Files. To adjust the reject reason you can use the option R - an appropriate EndToEnd ID can be e.g.:
    SCT Reject: **RJCT**-R:AM01
This will generate a reject for your transaction marked with that EndToEnd ID.

For SDD:
SDD-rejects have two different types of SEPA files - DVF and DNF - also the R-option can be supplied to adjust the reject reason. For example
    SDD Reject DVF: **RJTV**-R:AM01
    SDD Reject DNF: **RJTN**-R:AM01

For SCC:
SCC-rejects have two different types of SEPA files - DVF and DNF - also the R-option can be supplied to adjust the reject reason. For example
    SCC Reject DVF: **RJTV**-R:AM01
    SCC Reject DNF: **RJTN**-R:AM01

For SVV:
SVV-rejects are always resulting in DVF-Files. To adjust the reject reason you can use the option R - an appropriate EndToEnd ID can be e.g.:

    SVV Reject: `RJTV-R:AM05`

This will generate a reject for your transaction marked with that EndToEnd ID. For SVV, only the following reject reason codes are supported: AM05, DT01, ED05.

If the R-parameter is not provided AM05 will be set automatically.

5.2.4   Return (SCT, SDD, SCC, SVV, Recall)

PTS/SMV generates a Return message based on the original transaction (SCT, SDD, SCC, SVV, or Recall). The Return reason code is included in the command by using the [R-option](). EndToEnd IDs to generate Return messages for SCT, SDD, SCC or SVV are for example:

    SCT Return SCF: **RTRN**-R:MD07
    SDD Return SDF: **RTRN**-R:MD07
    SCC Return SDF: **RTRN**-R:MD07
    SVV Return SDF: **RTRN**-R:AC01
    Recall Return SCF: **RCR**-R:FOCR

For SVV, only the following return codes are supported: AC01, AC04, AG02, CUST, MS03.

If the R-parameter is not provided AC01 will be set automatically.

The Recall-Return can only be generated on Recall-Messages. To provide the EndToEnd ID, please use the original EndToEnd ID inside the original SCT. A recommended procedure is to build up a SCT having the EndToEnd ID on your SEPA-System. Then issue a Recall and send this recall to PTS/SMV in order to get the returned Recall.

5.2.5   Refund (SDD, SCC)

PTS/SMV generates a Refund message on the basis of the original transaction. The Refund reason code is included in the command.The handling procedure is same as [Return](). When PTS/SMV generates a refund, it always includes a generic compensation amount of 5 Euro.

Example:
    SDD Refund SDF: `RFND-R:MD06`
    SCC Refund SDF: `RFND-R:MD06`

If the R-parameter is not provided AC01 will be set automatically.

5.2.6   Reversal (SDD, SCC)

This function mirrors the original transaction and generates the Reversal. Again the Reversal reason code is included in the command. So, there will two files be generated – the SDD and the Reversal matching the new SDD. The EndToEnd ID may look like:

    SDD Reversal SDF: **RVSL**-R:MD01

SCC Reversal SDF: **RVSL**-R:MD01

If the R-option is not provided AM05 is used by default.

5.2.7 Request for Cancellation (SDD)

Similar to the [Reversal](#), this function generates the Request for Cancellation. So, there will be two files generated – the SDD and the Request for Cancellation matching the new SDD. The EndToEnd ID may look like:

SDD RfC DNF: **RQFC**-R:DUPL

If the R-parameter is not provided DUPL will be set automatically.

5.2.8 Recall (SCT, Recall)

In addition to mirroring the original transaction, this function generates the recall. Another function lets PTS/SMV send a NAK-message on an uploaded Recall.

Generating Recall:
This function will mirror the original CT-transaction and generates a Recall corresponding to this new message. Your original message is like the template to the new CT.
For example

SCT Recall SCF: **RCL**-R:DUPL

Generating Recall NAK:
This function works on Recalls you are uploading to PTS/SMV. This will generate a camt.029 message to simulate a negative feedback on the original Recall. As in camt.056 (Recalls) there is no EndToEnd ID, you have to add the original EndToEnd ID to the command.
For example

SCT Recall NAK SCF: **RCN**-AC04

If the R-parameter is not provided CUST will be set automatically.

Generating Recall NAK with multiple Additional Information tags:
This function works on Recalls with reason AC03 uploaded to PTS/SMV. This will generate a camt.029 message which is a negative feedback on the original Recall
with 1 to 11 additional information tags. None of the additional tags depend on EndToEndId.
For example

SCT Recall SCF: RCNAC03-R:AC04:5
this will generate SCF Camt.029 with 5 additional information tags.

5.2.9 Inquiries

In addition to mirroring the original transaction, this function generates the Inquiry.

- If ICF Pacs.008 is processed with <EndToEndId> as **CM27:BNR**, SMV mirrors SCF Pacs.008, OQF Camt.027.001.06

- If ICF Pacs.008 is processed with <EndToEndId> as **CM87:VDA**, SMV mirrors SCF Pacs.008, OQF Camt.087.001.06

5.2.9.1 ACK/NAK-message on an uploaded Inquiry

Another function lets PTS/SMV send an ACK/NAK-message on an uploaded Inquiry

- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:ACNR**, SMV simulates Positive OQF Camt.029.001.08
- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:ARDT**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:ARJT**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:NOOR**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:RR04**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.027.001.06 is processed with <EndToEndId> as **BNR:RNPR**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.087.001.06 is processed with <EndToEndId> as **VDA:ACVA**, SMV simulates Positive OQF Camt.029.001.08
- If IQF Camt.087.001.06 is processed with <EndToEndId> as **VDA:MODI**, SMV simulates Positive OQF Camt.029.001.08
- If IQF Camt.087.001.06 is processed with <EndToEndId> as **VDA:CVAA**, SMV simulates Negative OQF Camt.029.001.08
- If IQF Camt.087.001.06 is processed with <EndToEndId> as **VDA:RJVA**, SMV simulates Negative OQF Camt.029.001.08

5.2.9.2 SCT Inquiry Credit Transfer (Inter-PSP Fee and/or Compensation Payment)

This function triggers PTS/SMV to generate an SCF SCT Inquiry out of an uploaded ICF Normal SCT.

In order to enable this function, the EndToEnd ID used should be `INQSCT:FCOL-C:N,` where `FCOL` is one of the three Category Purpose codes allowed for pacs.008 with DS-11 and `N` is the amount that will be mapped to the Interbank Settlement Amount of the generated SCT Inquiry; and consequently also to its Total Interbank Settlement Amount.

- If an ICF pacs.008 with DS-02 is processed with <EndToEndId> as `INQSCT:FCOL-C:N`, then SMV generates an SCF pacs.008 with DS-11 for 'Fee Collection'.
- If an ICF pacs.008 with DS-02 is processed with <EndToEndId> as `INQSCT:INTE-C:N`, then SMV generates an SCF pacs.008 with DS-11 for 'Interest'.
- If an ICF pacs.008 with DS-02 is processed with <EndToEndId> as `INQSCT:FCIN-C:N`, then SMV generates an SCF pacs.008 with DS-11 for 'Fee Collection and Interest'.

5.3 Test Options

5.3.1 Error Code (R)

To specify reason codes for rejects, returns or refunds you can use the option *R*. PTS/SMV will not check the code used if it is valid (unless TESTMODE is set to "Yes" in the SEPADefinition configuration settings). This option is applicable to every function that generates any kind of R-transaction.

Examples:
```
RJCT-R:AM01
RTRN-R:MD07
```

5.3.2 Transactions in File (F)

This option can only be used when transactions are generated using the duplicate-function of PTS/SMV. It contains the maximum number of transactions for one file.

Examples:
```
ECHO-N:100-F:30
ECHO-N:1000000-F:99999
```

5.3.3 Duplication Count (N)

This option specifies the number of transactions to be generated for the duplicate function. If this option is used, the mirror function becomes the duplicate function.

Examples

5.3.4 EndToEnd ID (E2E)

This option can be used to specify the EndToEnd ID of a new transaction that is generated with the mirror function. Due to scheme limitations the value should be short enough so that the EndToEnd ID length is not exceeded. The value is restricted to alphanumeric values.

Examples:
```
ECHO-E2E:TESTCASE23
ECHO-E2E:BIC123TEST321
```

5.3.5 Return Charges (C)

This option specifies an amount to be returned as a charge on returns and refunds. It results in ChargesInformation-Tags allowing to test additional charges by other banks. This option is used for refunds and returns.

Examples:

```
RFND-R:DISP-C:5
RTRN-R:MD02-C:3
RCR-R:FOCR-C:3
```

5.4 Function List

PTS/SMV Function	Command	Option (mandatory / optional)	Remark
Mirror	ECHO	-E2E (o)	
Duplicate	ECHO	-N (o)	
		-F (o)	
Reject	RJCT	-R (m)	
	RJTV		
	RJTN		
Return	RTRN	-R (m)	
		-C (o)	
	RCR	-R (m)	RCR demands reason code FOCR
		-C (o)	
Refund	RFND	-R (m)	
		-C (o)	
Reversal	RVSL	-R (m)	
Request for Cancellation	RQFC	-R (m)	
Recall	RCL	-R (m)	
	RCN	-R (m)	

6 Functional Description for PAIN

PTS/SMV can also be used to test the PAIN processing. The following use cases are implemented.
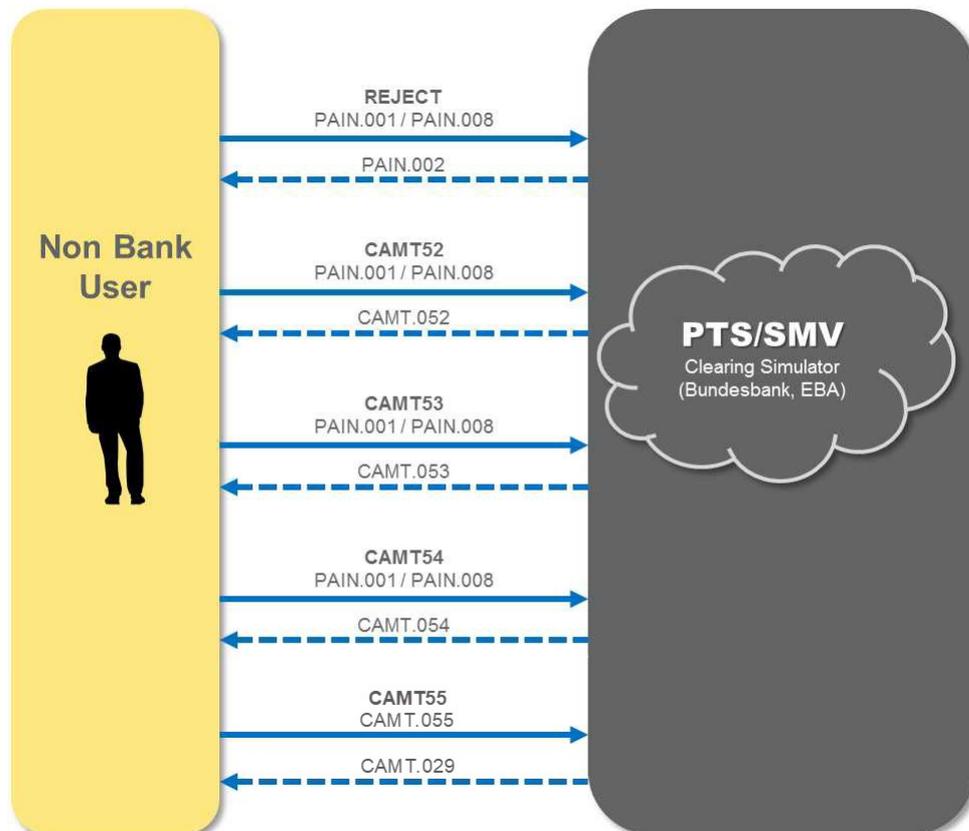
PAIN and CAMT Use Cases

The user (non bank) can hand in PAIN.001 or PAIN.008 files. The incoming files contain an action command - REJECT, CAMT52, CAMT53, CAMT54 – which determines the result file. The result file is filled with data of the incoming file as far as necessary the result file is also enriched with fantasy values.

The action command can be specified at file level concerning all transactions of the file or at transaction level concerning only certain transactions. In the first case the action command is specified within the `<MsgId>` element of the group header, in the second case within the `<EndToEndId>` of the transaction.

Cards payments are always processed as bulk. Therefore, the action command for CAMT should be inserted within `<MsgId>` (except for R-transactions).

If there is an action command at file (bulk) level all other contained action commands are simply ignored

If there is an action command at transaction level there may be options concerning the whole file. When there are for example two transactions with two different opening saldos then PTS/SMV just takes one of them.

The possible action commands are explained in the following chapters.

6.1   Test Functions PAIN

6.1.1   Action REJECT

The action command REJECT just rejects a PAIN.001 or PAIN.008 message. PTS/SMV creates a status report – a PAIN.002 message. The status report contains
- a message to the ordering party in case of a credit transfer

- a message to the receiver of the payment in case of a direct debit

The content of the fields of the status report is specified in the current version of *ZKA – DFÜ Abkommen Anlage 3*.

PTS/SMV supports the rejection of single transaction, of a bulk or of the whole file. The reason of the rejection should be specified.

Options
- *-R*   the reason of the rejection (if not set AC01 is used)
- *-B*   rejection on bulk level
  This option makes only sense if the action command is specified within the `<MsgId>` element, otherwise it is simply ignored. In case of the -B option every bulk within the file is rejected, otherwise the whole file.
- *-ID*   the original message ID or the original EndToEnd ID
  If the action command is specified within the `<MsgId>` element, then the ID is used as original message ID, if the action command is specified within the `<EndToEndId>` element of the transaction, then the ID is used as original EndToEndId.

Examples
- `<EndToEndId>REJECT-R:AC04</EndToEndId>`
  The transaction is being rejected. The status reason information is created on transaction level. The original EndToEndId `<OrgnlEndToEndId>` still contains the action command.

- `<EndToEndId>REJECT-ID:NewEndToEndId</EndToEndId>`
  The transaction is rejected. The status reason information is created on transaction level with the default value code AC01. The original EndToEndId `<OrgnlEndToEndId>` is replaced by NewEndToEndId.

- `<MsgId>REJECT-R:AC04-ID:NewMsgId</MsgId>`
  The whole file is being rejected. The status reason information is created on original group information and status `<OrgnlGrpInfAndSts>`. The original message ID is replaced by NewMsgId.

- `<MsgId>REJECT-R:AC04-B</MsgId>`
Because of the –B option (Bulk) the payment informations are being rejected. The status reason information is created on the original bulk element `<OrgnlPmtInfAndSts>`. The original messsage Id is not replaced, it still contains the command action.

6.1.2   Action CAMT 52

With the action command CAMT52 PTS/SMV creates an intraday account report. This action command is almost identical to the action command CAMT53.

Balances in CAMT52 are optional. So balances are only created when an opening saldo is specified.

6.1.3   Action CAMT 53

With the action command CAMT53 PTS/SMV creates an account report.

This report must contain saldo information (balance elements). Therefore an opening saldo is needed. If the opening saldo is not specified within the action command, PTS/SMV assumes an opening saldo of "0". The closing saldo is automatically calculated by PTS/SMV.

The structure of the account report also depends on the *BatchBooking* flag. But this flag is only considered when the action command is specified within the group header and the action command does not contain the "Return" option. In case of *BatchBooking=true* a report entry for the whole bulk (payment information) is created. The resolution of the transaction belonging to the bulk is within the report. PTS/SMV does not reference the details in an additional CAMT54 message.

Options (Optional)

- -ID   the original message ID or the original EndToEnd ID.
If the action command is specified within the `<MsgId>` element, then the ID is used as original message ID, if the action command is specified within the `<EndToEndId>` element of the transaction, then the ID is used as original EndToEndId.
- -O   the opening saldo
If not specified, PTS/SMV assumes „0". If the command is specified on transaction level, then the opening saldo should be the same within each EndToEndId-command. Otherwise PTS/SMV takes the first found opening saldo.
- -C   Charges in case of a return
Should be used only with the return option (`-R:<reason>`). Otherwise this option is ignored.
- -R   The reason
The transaction(s) is handled as a return, PTS cannot return a whole bulk. This option must provide a reason. It is not enough just to specifiy the option `-R`, but `-R:<reason>` i.e. `-R:AC01`.
- -S   Creates a summery
This is not part of *ZKA – DFÜ Abkommen Anlage 3* but it is often used by customers.

Examples

- `<EndToEndId>CAMT53-ID:NewId</EndToEndId>`
This action command in a pain.001 file creates an account report for the debtor of the incoming file. On the other hand this action in a pain.008 file creates an account report for the creditor of the incoming file. The EndToEndId with the action command is replaced by NewId. Because of no opening saldo (option -O) specified, a value of 0 € is assumed.

- `<EndToEndId>CAMT53-R:AC01-C:2</EndToEndId>`
This action command in a pain.001 file creates an account report for the debtor of the incoming file. On the other hand this action in a pain.008 file creates an account report for the creditor of the incoming file. The transaction is handled as R-transaction (return) with a charge of 2 €. The opening saldo is set to 0 €.

- `<MsgId>CAMT53-R:AC01</MsgId>`
This action command in a pain.001 file creates an account report for the debtor of the incoming file. On the other hand this action in a pain.008 file creates an account report for the creditor of the incoming file. All transaction within the pain file are returned with the reason AC01. The original message id is not replaced, the action command is still available.

6.1.4   Action CAMT 54

With the action command CAMT54 PTS/SMV creates an account report at the end of day or intraday.

In contrast to a [CAMT52](#) or [CAMT53](#) this report does not have saldo information, so there is no "-O" option (option for the opening saldo). Besides this option the command is identical to the action command [CAMT53](#).

6.1.5   Action CAMT 55

PTS/SMV supports (simulates) the case when the originator sends a CAMT.055.001.05 Request for Cancellation of SEPA Credit Transfer or SEPA Direct Debit to PTS/SMV with pre-defined command of what PTS/SMV should reply back:
positive CAMT.029.001.06 or negative CAMT.029.001.06.

Action command is CAMT55.
PTS/SMV supports the following use cases of CAMT.055:

1. Request for cancellation of a block of transactions. In this case an incoming CAMT.055.001.05 should contain:
    a. <CstmrPmtCxlReq><Undrlyg><OrgnlPmtInfAndCxl><**PmtInfCxl**> should equal to `true`
    b. <CstmrPmtCxlReq><Undrlyg><OrgnlPmtInfAndCxl><OrgnlGrpInf><**OrgnlMsgId** >**CAMT55-ST:XXXX-R:YYYY**, where
    XXXX is one of the available statuses {**CNCL**, **RJCR**, **PDCR**, **UWFW**, **CWFW**}
    YYYY - one of the reason or proprietary codes. Allowed codes for status RJCR are {**ARDT**, **NOOR**}, proprietary code is **MULT**.
    If for status RJCR a 4-letter code is provided and it is not {**ARDT**, **NOOR**, **MULT**}, then ERROR will be displayed.
    For other statuses the reason/proprietary codes are not mapped to CAMT.029 and

warning is displayed.

2. Request for cancellation of a transaction

3. Request for cancellation of several transactions

    For 2. and 3. an incoming CAMT.055.001.05 should contain per each transaction:
    a. <CstmrPmtCxlReq><Undrlyg><OrgnlPmtInfAndCxl><**PmtInfCxl**> should equal to `false`
    b. <CstmrPmtCxlReq><Undrlyg><OrgnlPmtInfAndCxl><TxInf><**OrgnlEndToEndId**
        >**CAMT55-ST:XXXX-R:YYYY**, where
        XXXX is one of the available statuses {**CNCL**, **RJCR**, **PDCR**, **UWFW**, **CWFW**}
        YYYY - one of the reason or proprietary codes. Allowed codes for status RJCR are
        {**CUST**, **AC04**, **AGNT**, **AM04**, **ARDT**, **LEGL**, **NOAS**, **NOOR**}, proprietary code is **MULT**.
        If for status RJCR a 4-letter code is provided and it is not {**CUST**, **AC04**, **AGNT**, **AM04**,
        ARDT, **LEGL**, **NOAS**, **NOOR**, **MULT**}, then proprietary code **MULT** will be mapped to
        CAMT.029.
        For other statuses the reason/proprietary codes are not mapped to CAMT.029 and a
        warning is displayed.

Available statuses are described in the table below:

PTS/SMV function	Format in CAMT.055	Command	ISO name	Case of usage	CAMT.029 positive ➕ CAMT.029 negative ➖
Cancel	CAMT55-ST:**CNCL**-	CNCL	CancelledAsPerRequest	Used when a requested cancellation is successful.	➕
Reject	CAMT55-ST:**RJCR**-	RJCR	RejectedCancellationRequest	Used when a requested cancellation has been rejected.	➖
Pending	CAMT55-ST:**PDCR**-	PDCR	PendingCancellationRequest	Can only be used with SCT. Used when a requested cancellation was transmitted to the receiver's ZDL, but the result is pending.	neutral
Pending	CAMT55-ST:**UWFW**-	UWFW	UnableToApplyWillFollow	Used when waiting for the original transaction. If the deadline is over the case will be terminated by an additional CAMT.029 via RJCR.	neutral
Pending	CAMT55-ST:**CWFW**-	CWFW	CancellationWillFollow	Cancellation request has already been recognized as technically and successfully viable - conduction / booking will follow.	neutral

6.1.6 Created Result Files

PTS/SMV generates multiple result files depending on the number of transactions within the incoming file and the contained action commands.

6.1.6.1 Action command on Bulk level (within MsgId of GroupHeader)

If an incoming pain file contains an action command within the group header, than normally one result file is created (if necessary one for each payment information). The structure of the result file may depend on the *BatchBooking* flag.

6.1.6.2 Action command on Transaction level (within EndToEndId)

If an incoming pain file contains one or more transactions with the same action command (for example all transactions have the action command CAMT54), then one result file is created (if necessary one for each payment information).

If an incoming pain files however, contains more than one transaction and these transactions have different action commands, then PTS/SMV will generate more than one result file.

7 Results

A report is generated for each SEPA file indicating possible errors and processing information. In addition, PTS/SMV can return other SEPA messages.

7.1 Reports

The report is provided by opening the result view of the corresponding original SEPA file. To do this, select your uploaded file in the file list by clicking it. Then press the information button. A View will open where you can view the report for your request and download generated files on demand. The report may look like:

Time	Message
2010-07-18 19:05:46.859	Succesfully validated
2010-07-18 19:05:49.451	found option in transaction (Generate Duplicates of Transaction) : 100
2010-07-18 19:05:49.451	found option in transaction (Use maximum of Transactions in File) : 30
2010-07-18 19:05:49.451	Transaction operation request: ECHO
2010-07-18 19:05:49.451	- Mirroring Transaction
2010-07-18 19:05:49.466	Generated file ECHO-100-FILE1-30TRANSACTIONS.xml
2010-07-18 19:05:49.466	Generated file ECHO-100-FILE2-30TRANSACTIONS.xml
2010-07-18 19:05:49.466	Generated file ECHO-100-FILE3-30TRANSACTIONS.xml
2010-07-18 19:05:49.466	Generated file ECHO-100-FILE4-10TRANSACTIONS.xml

7.2 Download

Within the same view, the user can download the generated new SEPA message files. Press the *Download* Button and find a *Zip Download* Dialog coming up. You can save this zip to your disk and unzip it to bring the files into your SEPA system.

8   Users

There are three different roles of users defined in PTS/SMV: superusers, admins and (non-admin) users.

- **Non-admin users** can register to PTS/SMV and allocate to a PTS/SMV cloud service account member (identified by its BIC). Alternatively they can be set up by an admin user.
- **Admin users** have to register as non-admin users first. Their role has to be changed to "admin" by a superuser. They can view all users of their own account member, register new ones or delete existing.
- **Superusers** are established at installation time. They can view all users of all account members and allocate roles (superuser, admin or user) to them.