Platform Documentation

XML Integrator (Transactions)

Please be aware of possible certification requirements for your platform (e.g. PCI DSS for credit card processing) if you decide for this integration option.

Version 3.4.4

Release History

Release	Description	Date	Changes
0.9	Beta Version	26-Nov-2003	Direct Debit and Credit Card XML
1.0.0	Initial Version	05-Jan-2004	Changed tag names, value formats and new Method, Type, Reason and Status Codes
[…]	[…]	[…]	[…]
2.0.6	Minor Version	28-Oct-2009	New Tag <Balance> for Loyalty Programs
3.0.0	Major Version	06-Jan-2011	Text revision and clarification
3.0.1	Minor Version	14-Mar-2011	Minor corrections
3.1.0	Minor Version	20-Apr-2010	Added UTF-8 encoding rules
3.1.1	Minor Version	09-Jun-2011	Minor corrections
3.1.2	Minor Version	26-Oct-2011	Minor corrections
3.1.3	Minor Version	02-Dec-2011	Minor corrections
3.2.0	Major Version	20-Feb-2012	SECURITYHASH generation via SHA-1 algorithm included
3.3.0	Major Version	30-Mar-2012	Airline Market Data available
3.3.1	Minor Version	11-Apr-2012	Minor corrections
3.3.2	Minor Version	09-Aug-2012	Minor corrections
3.3.3	Minor Version	13-Sept-2012	Minor corrections
3.3.4	Minor Version	20-Nov-2012	Registration details enhanced
3.3.5	Minor Version	08-Mar-2013	Minor corrections
3.3.6	Minor Version	21-Mar-2013	Minor corrections
3.4.0	Major Version	14-Aug-2013	Direct Debit SEPA specifics added, Virtual Account (Klarna Invoice) added
3.4.1	Minor Version	08-Oct-2013	Minor corrections
3.4.2	Minor Version	05-May-2014	Minor corrections
3.4.3	Minor Version	18-July-2014	Reviewed
3.4.4	Minor Version	29-May-2015	Minor correction

Content

1 XML Integrator Messaging

1.1 Introduction

The XML Integrator has two basic layers:

- Processing of XML requests and responses
- Transmission of XML files

The XML Integrator specification allows for processing of multiple payment methods. The following payment methods are currently supported:

Bank Transfer Methods

- Direct Debit DD
- Credit Transfer CT
- Prepayment PP
- Invoice IV

Card Payment Methods

- Credit Card CC
- Debit Card DC

Online Payment Methods

- Online Bank Transfer OT

Other Methods

- Virtual Account VA
- User Account UA
- Risk Management RM

In order to transmit, encrypt and decrypt a XML file the following options are currently available:

- HTTPS POST

1.2 General Structure

The complete XML Integrator specification is designed to be as easily understandable and polymorphic as possible. That means that all payment methods reuse the same tag set and coding scheme with very minor modifications.

1.3 Common ID and Coding Scheme

The XML Integrator has a common scheme for IDs and codes for all payment methods. There are three different types:

- Unique IDs
- IDs
- Codes

Unique IDs (UID) always have 32 alphanumeric digits and are used to uniquely identify the physical sender, the channel, the user login, a single transaction, a specific account or customer.

IDs have the format 0000.0000.0000 and are used to represent the short ID for a transaction (which are non unique).

Codes are used for payment or risk management methods, types, status and reason values. All codes are provided within the attribute 'code' of the specific tag. They have a hierarchical structure:

- Method code format: AA e.g. DD (Direct Debit) or RM (=Risk Mgmt.)
- Type code format: AA e.g. DB (Debit) or BL (=Blacklist)
- Status code format: 00 e.g. 90 (NEW) or 20 (=NEGATIVE)
- Reason code format: 00 e.g. 00 (Successf. P.) or 80 (Acct. in Blacklist)

In order to identify a specific payment or risk management type, the method and type code are concatenated:

- Payment / Validator (RM) codes have the format: AA.AA e.g. DD.DB or RM.BL

Every processing result is determined by the processing code and return code. The processing code is simply a concatenation of status and reason code:

- Processing codes have the format: 00.00 e.g. 90.00

The codes of very specific messages (which help to interpret a specific processing code) are provided within the return tag:

- Return codes have the format: 000.000.000 e.g. 100.100.400

1.4 Request Encoding

For all payment requests the request header must contain the Content-Type / charset parameter with the charset encoding set to "UTF-8". The actual content type may differ; the decisive information is the charset

value.

Accordingly, all request data must be encoded using the UTF-8 character set.

For XML data, please use the following Content-Type: application/x-www-form-urlencoded;**charset=UTF-8**

Examples:

Integration via PHP/cURL: http://php.net/manual/de/book.curl.php

```
$ch = curl_init();
curl_setopt($ch, CURLINFO_HEADER_OUT, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, array(
        "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"
    ));
// attach parameter
curl_exec($ch);
```

Integration via Java:

```
UrlRequest req = new UrlRequest(UrlRequest.POST, CORE_URL );
// attach parameter to request
req.addHeaderParam("Content-Type", "application/x-www-form-urlencoded;charset=UTF-8");
req.send();
```

1.5 Response Validation – HASH Digital Signature

To guarantee the integrity of the response message and parameters, a digital signature can optionally be generated by the system to confirm the integrity of the transaction. The receiver can then generate the same key in turn. The construction of the hash follows:

The SecurityHash verification value can be enabled via the BIP administration panel following the Administration – Processing – Processing Settings section.

The HASH tag is:

- labeled <SecurityHash>
- generated by SHA-1 encryption algorithm
- contains following parameters
  - PAYMENT.CODE
  - IDENTIFICATION.TRANSACTIONID
  - IDENTIFICATION.UNIQUEID
  - CLEARING.AMOUNT
  - CLEARING.CURRENCY
  - PROCESSING.RISK_SCORE
  - TRANSACTION.MODE
  - PROCESSING.RETURN.CODE
  - PROCESSING.REASON.CODE
  - PROCESSING.STATUS.CODE
  - and the "secret" generated during the Merchant Account setup; only known by the merchant
- not placed in a group, but counts as a global parameter for verification purposes

The String to generate the hash is constructed as follows:
PAYMENT.CODE + "|" + IDENTIFICATION.TRANSACTIONID + "|" + IDENTIFICATION.UNIQUEID + "|" + CLEARING.AMOUNT + "|" + CLEARING.CURRENCY + "|" + PROCESSING.RISK_SCORE + "|" +

TRANSACTION.MODE + "|" + PROCESSING.RETURN.CODE + "|" + PROCESSING.REASON.CODE + "|" + PROCESSING.STATUS.CODE + "|" + secret

Note: If one of the parameters above is not part of the response message, the parameter is set to an empty string like the "PROCESSING.RISK_SCORE" in the following example (bold) "CC.DB|MerchantAssignedID|402880e5faf35d0700faf35d0cec0002|298.00|EUR**||**INTEGRATOR_TEST|000.100.110|00|90|abcd1234".

1.6 Request Sample

Each request contains a Header tag with transmission and security related information and one Transaction tag.

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="DD.DB">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
         <Country>DE</Country>
      </Account>
      <Customer>
         <Name>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
   </Transaction>
</Request>
```

1.7 Response Sample

The XML response repeats some information of the request and adds additional tags to it. The most important additions are the UniqueID, ShortID and the Processing group.

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

1.8 Methods and Types

Payment Methods and Types are the most central elements of a request, because they determine the behavior and mandatory elements of the request.

In order to build or parse a specific method or type code, there is an easy scheme for all occurrences. All type codes come from the same general set of codes and are polymorphic. For example, this is a method and type pairing of a direct debit refund:

```
<Payment code="DD.RF">
```

which is the same for a credit card (CC) transaction (bold, underline) of the refund (RF) type - except the preceding method code:

```
<Payment code="CC.RF">
```

The complete set of method codes is listed below. More information is available on the wiki or via an account manager.

Method	Code	Description
Bank Transfer Methods		
Direct Debit	DD	The merchant collects money from the shopper account. This generally requires the approval by the shopper.
Credit Transfer	CT	The merchant transfers money to the bank account of the customer or to any other bank account. This method is mainly used for payout

		or cash management purposes.
Prepayment	PP	The shopper receives all necessary credit transfer details and transfers money to the bank account of the merchant. The incoming payment gets matched and subsequently the goods or services can be shipped to the shopper.
Invoice	IV	Similar to Prepayment but with the difference that the goods or services are delivered to the shopper before the money receipt. This is common in the B2B sector.

Card Payment Methods		
Credit Card	CC	The credit card of the shopper gets debited with the payment amount. In regular intervals the shopper receives an invoice from the issuing institute of the credit card.
Debit Card	DC	The debit card of the shopper gets debited with the payment amount. The customer must have deposited money on his debit card or his bank account before the transaction can get authorized.
Online Payment Methods		
Online Transfer	OT	The shopper performs an online credit transfer with the homebanking interface of his bank. The merchant receives an acknowledgement as soon as the authorisation of the credit transfer was successful. Examples are IDEAL (NL), Giropay (DE) or EPS (AT).
Other Methods		
User Account	UA	A registered user in the system. Typically a Virtual Account and other payment methods will be attached to it.
Virtual Account	VA	The shopper credits a virtual account within the merchant shop and sequentially debits this account. The account can be credited with all above payment methods and typically gets debited with micro payment amounts.
Risk Management	RM	If certain Risk Management checks shall be performed without a relation to a real payment transaction, the prefix RM can be used.

All payment methods use a subset of 8 available payment types. The naming of the standard payment types always references the action taken against the account of the shopper. The Payment Types are defined as follows:

	Merchant Account	
	Debit	Credit
Cash Flow Neutral	Preauthorisation (PA)	
Standard Types	Credit (CD)	Debit (DB)
	Refund (RF)	Rebill (RB)
Reverses Standard Types	Reversal (RV)	

Back Channel Types	Chargeback (CB)	Receipt (RC)

A detailed description of the behavior of these payment types plus all recurring and risk management types can be found in the table below:

Type	Code	Description
Payment Types		
Preauthorisation	PA	Performs all risk management validations and stores the complete payment processing data for later retrieval. For credit card transactions the payment amount is additionally reserved on the credit card of the shopper (in case the underlying acquirer supports this functionality). A subsequent Capture (CP) with reference to the Preauthorisation (PA) will trigger the actual booking.
Debit	DB	Debits the account of the shopper and credits the account of the merchant. This payment type combines a PA and CP into one call, rather than a "dual message" capture of funds.
Capture	CP	Captures an already preauthorized (PA) amount.
Credit	CD	Credits the account of the shopper and debits the account of the merchant.
Reversal	RV	Reverses an already processed Debit (DB) or Credit (CD) transaction. As consequence the shopper will never see any booking on his statement. A Reversal is only possible until a connector specific cut-off time applies. Some connectors don't support Reversals.
Refund	RF	Credits the account of the shopper with a reference to a prior Debit (DB) or Credit (CD) transaction. This can be done in order to refund an already processed Debit transaction or credit again an already processed Credit transaction. The shopper will always see two bookings on his statement. Refunds are supported by many (but not all!) connectors.
Rebill	RB	Debits the account of the shopper with a reference to a prior Debit (DB) transaction. This is normally used to rebill an already processed Debit (DB) transaction in case of a chargeback or to add additional products to an already processed order.
Chargeback	CB	A negative booking on the merchant account which is generally triggered by a return of a Debit (DB) or Rebill (RB) transaction by the shopper/bank.
Receipt	RC	A positive booking on the merchant account which is generally triggered by a return of a Credit (CD) or Refund (RF) transaction by the bank or by a credit transfer of an shopper to the merchant's bank account. This kind of receipt is used especially in conjunction with prepayment or invoice transactions.
Registration Types		
Registration	RG	Initial registration of an account and/or customer. Subsequent Debit (DB) or Credit (CD) transactions can always reference to a valid registration. By using this process no sensitive shopper account information needs to be stored within the merchant system. A registration can optionally also contain recurring rules in order to generate future Debit (DB) or Credit (CD) transactions automatically. For some countries a PDF mandate is produced as part of the direct debit registration process.

Reregistration	RR	If the registration data of an account or customer is changing, a reregistration of this data can be performed.
Deregistration	DR	In order to deactivate a prior registration, a deregistration transaction can be submitted to the system. All following Debit (DB) or Credit (CD) transactions which reference to the registration will be rejected as soon as the deregistration has been acknowledged.
Confirmation	CF	If the registration is used for the automatic creation of PDF direct debit mandates, the receipt of an signed copy of such a mandate 'confirms' the registration. This Confirmation (CF) transaction is either generated automatically by the system or by a manual entry process.

Scheduling Types		
Schedule	SD	Schedules upcoming payment transactions of the same type and amount. Generally one or more SD transactions are used to define a subscription pattern. The exact execution parameters and cancellation notice restrictions are provided within the Job tag group.
Reschedule	RS	Reschedules a scheduled job while the job is already running. Generally used to change the execution or cancellation notice parameters of a specific subscription.
Deschedule	DS	Cancels a running job. The cancellation notice does take effect after the period defined within the Notice tag has elapsed.

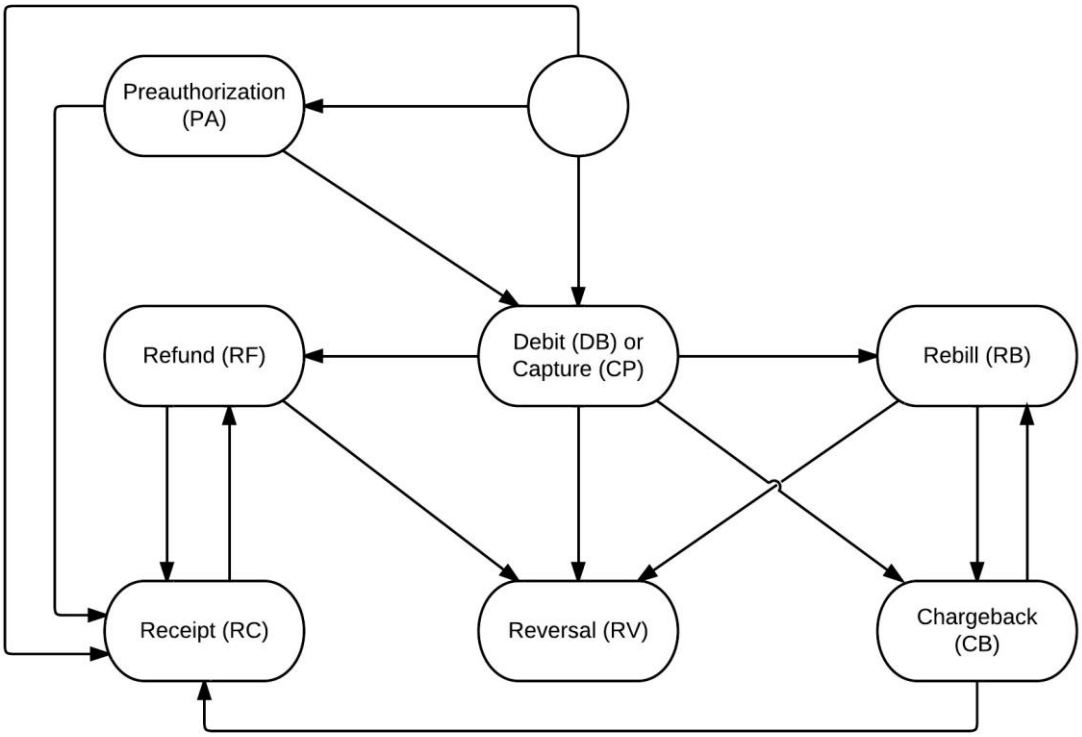
1.9 Type Logic

The below type logic diagrams indicate possible workflows for each of the types above.

There are two primary actions: the Debit logic and the Credit logic, dependent on whether the shopper account is to be debited or credited. The following diagrams note these logics with the following color coding:

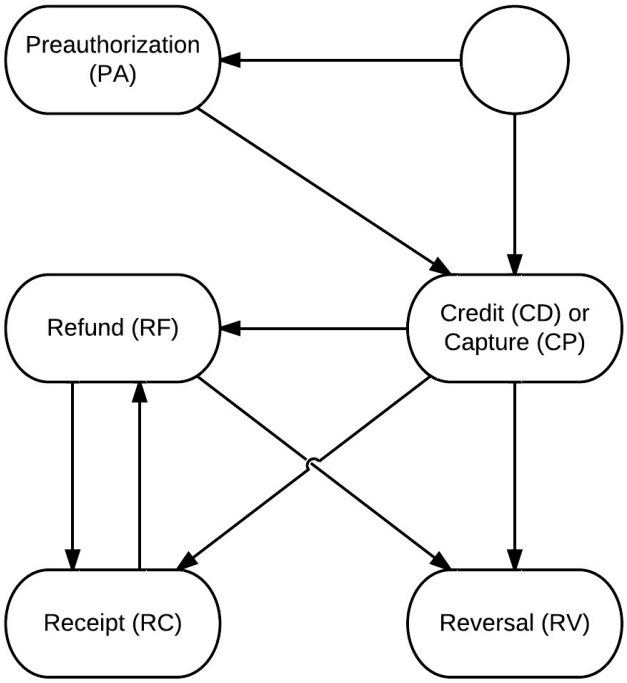
Black: Cash flow positive transaction type
Red: Cash flow negative transaction type
Grey: Cash flow neutral transaction type

1.9.1 Debit Logic



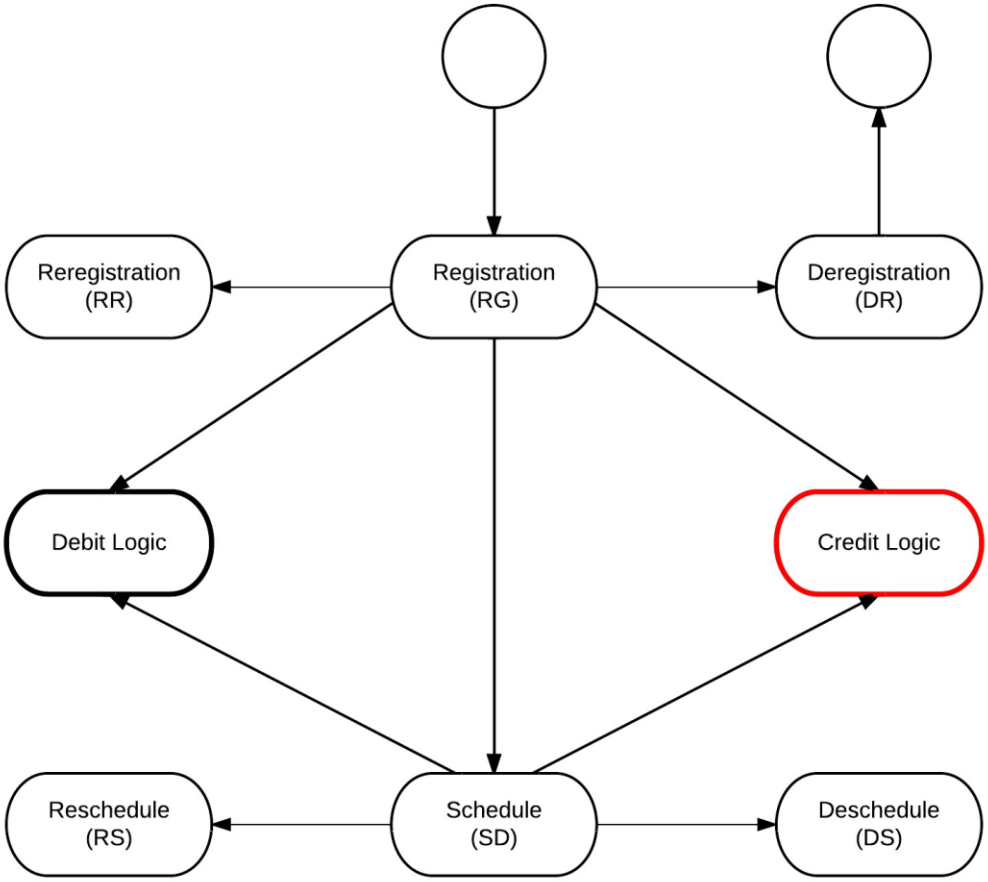
1.9.2 Credit Logic

The credit logic is a little bit simpler since no Rebill (RB) or Chargeback (CB) transactions can occur.



If payments are executed in a repeated manner, the recurrence logic applies. All customer and account data of recurrent transactions need to be registered (RG) initially to be available for later credit or debit logic transactions. This is also known as "tokenized" payments. Generally recurring transactions can be either triggered on demand or scheduled (SD) automatically to generate transactions against a specific schedule of payments.

1.9.3 Recurrence Logic


Reregistration
(RR)
Registration
(RG)
Deregistration
(DR)
Debit Logic
Credit Logic
Reschedule
(RS)
Schedule
(SD)
Deschedule
(DS)

1.10 Statuses and Reasons

The below-listed status codes are returned in the response of a transaction. They are accompanied with a reason code further explaining the high-level status.

Status	Code
NEW	90
WAITING	80
REJECTED_VALIDATION	70
REJECTED_RISK	65
REJECTED_BANK	60
NEUTRAL	40
SUCCESS	00

The combination of status and reason code is always unique and is generally used for more than one payment or risk management type. For a complete and current list of all possible type, status and reason codes please refer to https://test.ctpe.io/payment/codes/statusCodes.jsp

1.11 Header

The header group of the XML file holds transmission- and security related information.

```
<Header>
   <Security sender="123a456b789c123d456e789f012g345"/>
</Header>
```

Value for sender	Description
Alphanumeric 32	Each entity (PSP, Division, Merchant, Channel) which sends requests to the system has its own sender unique ID. The sender UID is no logical business orientated subdivision like the channel ID, but refers to physical installations of software. Please provide the value you have received from your account manager here.

1.12 Transactions

A request or response message can contain one or more 'Transaction' tags. The Transaction tag and its nested sub tags contain all information required to process a transaction.

```
<Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
```

The Transaction tag has three attributes that determine the processing of the transaction.

Value for mode	Description
INTEGRATOR_TEST	Transaction is sent to the system and is processed internally via the system's simulators; the Validator (Risk Management) or Connector modules are not called.
CONNECTOR_TEST	Transaction enters the Integrator module, accesses the Validator modules (Risk Management) and then goes to the Connector's test environment itself.
LIVE	Transaction enters the Integrator module, accesses the Validator modules (Risk Management) and then goes to the Connector. The Connector operates in live mode.
Value for response	**Description**
SYNC	Transaction gets processed in the synchronous mode, which means the client will get the result of the processing in the transaction's response.
ASYNC	Transaction will be processed in the asynchronous mode. Typical examples are asynchronous processes like 3DSecure or Online Bank Transfer. The direct response will just acknowledge the receipt of the transaction for processing. The result of the processing gets delivered back as an indirect response over another asynchronous communication channel like a SFTP repository or a POST call back method.
Value for channel	**Description**
Alphanumeric 32	The channel ID is a unique key for the identification of the unit that sends transactions into the system. Every merchant can have multiple channels for different purposes. Possible division criteria are e.g. different shops, organizational units, customer groups or countries. The channel ID does *not* refer to physical installations of the software like the sender ID but is a logical business orientated subdivision. Different channels help to analyze the entirety of transactions and to provide different system configurations for a non-uniform transaction base. Channel IDs are assigned by account management.

1.13 User Group

Every request submitted to the system needs to contain the User group that allows identification of a particular sender of the respective request.

For all automated requests:
A default system user gets created and assigned to every merchant automatically. The login credentials for this default system user can be retrieved from the Business Intelligence Platform (BIP). For all automated

requests, these login credentials must be used. In case of any questions please check with your account manager.

For manual transactions:
For eTerminal transactions or transactions triggered manually in the BIP (e.g. Refunds or Rebills) every regular system user can use its login credentials. (Please do not use these personalized login credentials for automated requests!)

Example for the login credentials:

```
<User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
```

Value for login	Description
Alphanumeric 32	The login is a unique ID for each human or system user. Each merchant or payment service provider can have several logins for system users and human users. It is *not* recommended to share one login between several human users.
Value for pwd	**Description**
Alphanumeric 5..32	The password matched to the login parameter.

1.14 Identification Group

The identification group contains IDs that are used for the identification of the transaction:

- Transaction ID
- Reference ID
- Unique ID
- Short ID
- Shopper ID
- Invoice ID
- Bulk ID

In the request, the merchant can provide a Transaction ID for own matching purposes. For transaction types that require a reference to a former transaction, (eg Capture, Reversal, Refund, Reregistration, Deregistraiton, Reschedule) a Reference ID has to be provided. The Shopper ID is used to group transactions of a certain shopper.

Request:

```
<Identification>
   <TransactionID>MerchantAssignedID</TransactionID>
   <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
   <ShopperID>customerid123456</ShopperID>
   <InvoiceID>2009010023</InvoiceID>
   <BulkdID>Batch12</BulkID>
</Identification>
```

During processing the system generates a universal unique ID (UniqueID). This Unique ID must be used for all automated matching and search purposes.

The Reference ID therefore is the Unique ID of another transaction to be referenced. ex: captures, reversals and refunds of an originating transaction.

To provide an ID that is shorter and easier to enter manually, a Short ID is also provided. The Short ID is used for the descriptor of the transaction and to search manually for transactions in the business intelligence platform. The Short ID doesn't guarantee universal uniqueness; however, the probability for non-uniqueness (especially in the context it appears) is very low.

Response:

```
<Identification>
   <TransactionID>MerchantAssignedID</TransactionID>
   <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
   <ShortID>1234.5678.9876</ShortID>
   <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
   <ShopperID>customerid123456</ShopperID>
   <InvoiceID>2009010023</InvoiceID>
   <BulkdID>Batch12</BulkID>
</Identification>
```

Tag of Identification	Data Type	Length	Mandatory / Optional	Description
UniqueID	Alphanumeric	32	Part of the Response, not part of the Request	Only ID where the uniqueness within the system is absolutely guaranteed. Has to be used for all automated matching and reference purposes.
ShortID	Numeric / Dots	14	Part of the Response, not part of the Request	ID that is used for manual entry and search purposes. The likelihood for uniqueness is very high, but not guaranteed.
TransactionID	Alphanumeric	0..256	Optional	ID assigned by the merchant. Uniqueness is *not* enforced by the system across any level of the system's hierarchy.
InvoiceID	Alphanumeric	0..256	Optional	ID assigned by the merchant to assign a transaction to a certain invoice. Typically this invoice ID is the ID the merchant also communicates to the shopper for a certain invoice. (ex: a purchase order)
ShopperID	Alphanumeric	0..256	Optional	ID assigned by the merchant to assign a transaction to a certain shopper. Typically the user ID or customer ID of the shopper within the merchant's shop system provided here. It can be used to search for all transactions of one shopper in the analysis backend tool.
ReferenceID	Alphanumeric	32	Cond. Mandatory	References the Unique ID of another transaction. Only needed for the submission of following transaction types: Capture, Reversal and Refund. Chargeback and Deposit transactions contain the Reference ID in the response message.
BulkID	Alphanumeric	0..64	Optional	ID assigned by the merchant. It can be used to mark transactions with a common field which logically belong together, e.g. all transactions sent in on a given date

				by sending in the date as the bulk id.

1.15 Payment Group

The Payment group determines which payment method and type to use; it also provides all monetary payment details of the transaction. Furthermore it contains the description of the transaction by means of the usage and descriptor tags.

```
<Payment code="DD.DB">
      ...
</Payment>
```

If you want to use certain payment methods, please ensure that the applied Channel ID has been activated for these methods by the account management.

Depending on the chosen payment method, there are specific types available:

		Bank Transfer				*Card Payment*		*Other*		
Payment Types		DD	CT	PP	IV	CC	DC	OT	VA	RM
Preauthorisation	PA	x	x	x	x	x	x	x	x	x
Debit	DB	x				x	x	x	x	
Credit	CD		x						x	
Reversal	RV	x	x			x	x		x	
Refund	RF	x	x	x	x	x	x	x	x*	
Rebill	RB	x				(x)	(x)			
Chargeback	CB	x				x		x		
Receipt	RC	x	x	x	x			x		
Registration Types										
Registration	RG	x	x	x	x	x	x	x	x	
Confirmation	CF	x				x	x		x	
Reregistration	RR	x	x	x	x	x	x	x	x	
Deregistration	DR	x	x	x	x	x	x	x	x	

* Applies to certain virtual account brands (such as PayPal) only. Please check with your Account Manager regarding which VA payment brands do and do not support refunds.

A detailed description of the payment types can be found in the respective chapters of this document. The

following table provides a basic overview of some very common payment types:

Type Name	Description
Debit	Debits the payment method of the shopper. (cash-flow positive)
Credit	Credits the bank account of the shopper. (cash-flow negative)
Preauthorisation	Reserves a certain amount on the payment method of the shopper. This amount can be captured later. Certain Acquirers call this transaction type also reservation. (cash-flow neutral)
Reversal	Reverses the transaction before the clearing cut off time of the processor or bank. A reversed transaction does not appear on the statement of the shopper. (cash-flow negative)
Refund	After the cut off time has expired, executing a Refund transaction is the only way to refund a customer. The amount of the Refund transaction can also be less than the original amount (aka: partial refund). (cash-flow negative)
Chargeback	If a debit transaction of a credit card or direct debit account is returned for any reason, a Chargeback is loaded into the system. (cash-flow negative) *Note: Not all processors / acquirers provide chargeback feeds. Please reach out to your account manager for more information regarding chargeback reporting.*
Receipt	If a credit transfer transaction is returned by a bank, or if the shopper makes a payment to the clearing account, a Receipt is created. (cash-flow positive)

In the request, the Payment group must include the Presentation tag. The response then returns the Clearing tag. The Clearing tag contains all data as the shopper would see on his or her bank statement.

Request:

```
<Payment code="DD.DB">
   <Presentation>
      <DueDate>2013-08-13</DueDate>
      <Mandate id="123456" dateOfSignature="2013-08-01"/>
      <Amount>1.00</Amount>
      <Currency>EUR</Currency>
      <Usage>Order Number 1234</Usage>
   </Presentation>
</Payment>
```

Response:

```
<Payment code="DD.DB">
   <Presentation>
          <DueDate>2013-08-13</DueDate>
          <Mandate id="123456" dateOfSignature="2013-08-01"/>
       </Presentation>
   <Clearing>
      <Amount>0.50</Amount>
      <Currency>GBP</Currency>
      <Descriptor>shop.de 1234.5678.9876 +49 (89) 1234 567 Order Number
1234</Descriptor>
```

```
        <FxRate>0.500000</FxRate>
        <FxSource>FTX</FxSource>
        <FxDate>2003-02-12</FxDate>
    </Clearing>
</Payment>
```

All tags within the Presentation tag are mandatory:

Tags of Presentation	Data Type	Length	Mandatory / Optional	Description
Amount	#0.00	1..10.2	Mandatory	Presentation Amount in currency of the Currency tag. *A dot is used as decimal separator, no separators are needed for 1000s etc.*
Currency	Alpha	3	Mandatory	Currency code according to the ISO 4217 specification plus the currency "PTS" for Points (e.g. used for loyalty programs). The complete list of supported currencies is available for download here: https://test.ctpe.io/payment/codes/currency.jsp
Usage	Alphanumeric	0..128	Mandatory	Provides the dynamic part of the descriptor, which appears on the shopper's statement. Enables the shopper to associate the transaction on the statement to the online transaction.

The BankName tag only appears for direct debit or credit transfer transactions in countries where the name of the clearing bank can be retrieved. The Fx ... tags within the Clearing tag are only returned if a currency conversion had to take place.

Tags of Clearing	Data Type	Length	Mandatory / Optional	Description
Amount	#0.00	1..10,2	Mandatory	Settlement amount on the shopper's account in currency of the Currency tag. *A dot is used as decimal separator, no separators are needed for 1000s etc.*
Currency	Alpha	3	Mandatory	Currency code according to the ISO 4217 specification plus the currency "PTS" for Points (e.g. used for loyalty programs). The complete list of supported currencies is available for download here: https://test.ctpe.io/payment/codes/currency.jsp
Descriptor	Alphanumeric	14..256	Mandatory	Appears on the statement of the shopper. The Descriptor is a concatenation of the Short ID, system defined text and the Usage tag. Enables the shopper to associate the transaction on his or her statement with the online transaction from the gateway.
DueDate	Date (yyyy-mm-dd)	10	Cond. Mandatory	Required for SEPA DD. If not provided, the value is calculated accordingly to the acquirer bank and SEPA scheme rules. By default the shortest possible due date period is calculated: 5 days for one-off payments, 2 days for recurring payments, 1 day for all kind of payment with COR1 format. All calculations are based on target-1 calendar and bank days.
Mandate id	Alphanumeric	1….256	Cond. Mandatory	Required for SEPA DD as an arbitrary mandate (document) identifier. If not provided, unique ID of first associated DB or RG transaction will be used. Merchant needs to store provided mandate id for later reference.
dateOfSignature	Date (yyyy-mm-dd)	10	Cond. Mandatory	Date of the mandate (document) signature required for SEPA DD. If not provided, date of the first associated DB or RG transaction will be used.
FxRate	*#0.000000*	*1..6,6*	*Cond. Mandatory*	*If the currency provided in presentation tag is not equal to the currency of the shopper's account, Direct Debit and Credit Transfer transactions have to be converted. The applied Foreign Exchange Rate is shown with a precision of 6 decimal places after the dot.*
FxSource	*Alphanumeric*	*0..256*	*Cond. Mandatory*	*The Source of the applied Foreign Exchange Rate.*
FxDate	*Date (yyyy-*	*10*	*Cond.*	*The applied rates are fixed rates of the applied*

	mm-dd)		*Mandatory*	*Foreign Exchange Source for a certain date.*
Balance	#0.00	1..10,2	Optional	Returns the remaining balance on the account. This field is only set for accounts (e.g. loyalty cards) that support this feature.
Value Date	Date (yyyy-mm-dd)	10	Cond. Mandatory	Date and time of the purchase.

Please note that in case of a chargeback the presentation amount is the amount returned by the bank and *not* by the merchant.

1.16 Recurrence Group (Manual Recurrence)

Recurring transactions are flagged with the <Recurrence> tag.

For the initial transaction (containing the CVV code) the following XML tag needs to be submitted:

```
<Recurrence mode="INITIAL"/>
```

All recurring transactions (without a CVV code) need to be submitted with this tag:

```
<Recurrence mode="REPEATED"/>
```

Tags and Attributes	Data Type	Length	Mandatory/ Optional	Description
@mode	Alphanumeric	0..64	Mandatory	One of "NONE", "INITIAL" or "REPEATED"

This tag will only take effect if the used Merchant Account is properly configured for recurring transactions in the BIP.

Please check with your account manager if for availability of recurring transactions.

1.17 Job Group (Automatic Recurrence)

The scheduling logic is generally used for payments that are executed in a repeated manner like subscriptions or regular payouts. See chapter 7 Scheduling Types (Automatic Recurrence) for an overview of typical Scheduling requests and responses. The document "Recurrence Scenarios" shows real life examples and usages of recurrences and subscriptions.

The Job group contains all essential details that describe the contractual parameters between merchant and customer.

With this functionality, a payment can be repeated or "scheduled" for multiple transactions. For such a use

case, the Schedule (SD) type needs to be set as the payment code:

```
<Payment code="CC.SD">
```

In this case a job is scheduled which automatically generates recurrent transactions with the data provided within the Payment tag. The execution and cancellation notice rules also can be provided within the Job tag.

The following example describes a Job group which can be used for a 3 months subscription with the following details: on every 13th of a month a Debit (DB) a transaction with the presentation amount and usage is triggered. The cancellation notice is 3 days until the end of the subscription period. If no cancellation has taken place, the subscription period restarts for another 3 months.

```
<Payment code="DD.SD">
   <Presentation>
      <Amount>19.90</Amount>
      <Currency>EUR</Currency>
      <Usage>Standard Subscription</Usage>
   </Presentation>
</Payment>
<Job name="Standard Monthly Subscripton">
   <Action type="DB"/>
   <Execution>
      <DayOfMonth>13</DayOfMonth>
      <Month>*</Month>
   </Execution>
   <Notice>
      <Number>3</Number>
      <Unit>DAY</Unit>
   </Notice>
   <Duration>
      <Number>3</Number>
      <Unit>MONTH</Unit>
   </Duration>
</Job>
    <Account registration="678a456b789c123d456e789f012g432"/>
```

If a scheduled payment needs to be canceled, a transaction of the "Deschedule (DS) type needs to be triggered, which stops the scheduled recurring process at the next possible cancellation point. Please refer to chapter 7 “Scheduling” and the document “Recurring Scenarios” for more details.

If a recurrence scenario contains transactions with different payment amounts or different contents of the Payment tag group, several Schedule transactions can be chained which allows modeling very advanced recurrence scenarios. Ex: first month free, latter 3 months at ½ price.

Please refer to the document “Recurrence Scenarios” to get numerous recurrence examples and how to model them with the provided framework.

Some virtual account / asynchronous providers also support scheduling mechanisms. Please check the document “Asynchronous Workflows” for more details.

Tags and Attributes	Data Type	Length	Mandatory / Optional	Description
Job/@name	Alphanumeric	0..64	Mandatory	Name of the Job. Used for identification purposes within the Business Intelligence Platform.
Job/@start	Timestamp	19	Optional	The request timestamp of the transaction is the default starting point of the recurring period. If a later starting point is desired, a different timestamp can be provided within the start attribute. The start attribute must be placed before the first desired execution time! WANRING: If you provide a start date in the past the job might fire (only ONCE) immediately depending on your <Execution> settings.
Job/@end	Timestamp	19	Optional	Timestamp that indicates the absolute end of the job. If no end attribute is provided the Job automatically restarts after the end of the period defined within the Duration tag. If no automatic prolongation is desired the end attribute should be equal to the timestamp after the subscription duration.
Action/@type	PA, DB, CD	2	Mandatory	Defines the type of the action that is executed periodically by the job. If the payment code of the scheduled transactions shall be e.g. CC.DB, the payment code of the Scheduling transaction has to be CC.SD and the type attribute of the Action tag 'DB'.
Execution/Second	0-59 , - * /	1..64	Optional	The default value is '0'. Please see below for an explanation of the quoted special characters.
Execution/Minute	0-59 , - * /	1..64	Optional	The default value is '0'
Execution/Hour	0-23 , - * /	1..64	Optional	The default value is '0'
Execution/ DayOfMonth	1-31 , - * ? / L W C	1..64	Optional	The default value is '1'. Be aware to use L instead of 29, 30 or 31 if you want to fire every month (as not every month has 31 days!)
Execution/Month	1-12 or JAN-DEC	1..64	Optional	The default value is '1'

	, - * /			
Execution/ DayOfWeek	1-7 or SUN-SAT , - * ? / L C #	1..64	Optional	The default value is '1'
Execution/Year	empty, 1970-2099 , - * /	1..64	Optional	The default value is '*'
Execution/ Expression	Alphanumeric	1..256	Optional	All values mentioned above can optionally also be provided as a Cron-Expressions. A Cron-Expression is a string comprised of the 7 fields mentioned above, while the Year field can be left out. All fields are separated by white space. See the end of this chapter for Cron-Expression examples.
Duration/Number	Alpha	1..16	Optional	The duration of the job period. As soon as the job period is elapsed and no cancellation has taken place or end attribute does take effect, the job is restarted.
Duration/Unit	HOUR, DAY, WEEK, MONTH, YEAR		Optional	The job period can be defined within different units.
Notice/Callable	ANYTIME, DURATION_END		Optional	Defines the reference point for the cancellation notice. Either the notice period references to the end of the duration or the notice is callable anytime (this is the default). If the duration is DURATON_END, Duration Number and Unit must be specified.
Notice/Number	Alpha	1..16	Optional	The Notice group defines how much time in advance a cancellation notice has to be given in order to avoid a prolongation of the job period.
Notice/Unit	HOUR, DAY, WEEK, MONTH, YEAR		Optional	Unit within the notice period is defined.

The special characters that are possible within the Execution tags have the following meaning:

Character	Description
*	is used to specify all values. For example, "*" in the minute field means "every minute".
?	is allowed for the day-of-month and day-of-week fields. It is used to specify 'no specific value'. This is useful when you need to specify something in one of the two fields, but not

	the other. See the examples below for clarification.
-	is used to specify ranges. For example "10-12" in the hour field means "the hours 10, 11 and 12".
,	is used to specify additional values. For example "MON,WED,FRI" in the day-of-week field means "the days Monday, Wednesday, and Friday".
/	is used to specify increments. For example "0/15" in the seconds field means "the seconds 0, 15, 30, and 45". And "5/15" in the second field means "the seconds 5, 20, 35, and 50". You can also specify '/' after the '*' character - in this case '*' is equivalent to having '0' before the '/'.
L	is allowed for the day-of-month and day-of-week fields. This character is short-hand for "last", but it has different meaning in each of the two fields. For example, the value "L" in the day-of-month field means "the last day of the month" - day 31 for January, day 28 for February on non-leap years. If used in the day-of-week field by itself, it simply means "7" or "SAT". But if used in the day-of-week field after another value, it means "the last xxx day of the month" - for example "6L" means "the last Friday of the month". When using the 'L' option, it is important not to specify lists, or ranges of values, as you'll get confusing results.
W	is allowed for the day-of-month field. This character is used to specify the weekday (Monday-Friday) nearest the given day. As an example, if you were to specify "15W" as the value for the day-of-month field, the meaning is: "the nearest weekday to the 15th of the month". So if the 15th is a Saturday, the trigger will fire on Friday the 14th. If the 15th is a Sunday, the trigger will fire on Monday the 16th. If the 15th is a Tuesday, then it will fire on Tuesday the 15th. However if you specify "1W" as the value for day-of-month, and the 1st is a Saturday, the trigger will fire on Monday the 3rd, as it will not 'jump' over the boundary of a month's days. The 'W' character can only be specified when the day-of-month is a single day, not a range or list of days.
LW	The 'L' and 'W' characters can also be combined for the day-of-month expression to yield 'LW', which translates to "last weekday of the month".
#	is allowed for the day-of-week field. This character is used to specify "the nth" XXX day of the month. For example, the value of "6#3" in the day-of-week field means the third Friday of the month (day 6 = Friday and "#3" = the 3rd one in the month). Other examples: "2#1" = the first Monday of the month and "4#5" = the fifth Wednesday of the month. Note that if you specify "#5" and there are not 5 of the given day-of-week in the month, then no firing will occur that month.
C	is allowed for the day-of-month and day-of-week fields. This character is short-hand for "calendar". This means values are calculated against the associated calendar, if any. If no calendar is associated, then it is equivalent to having an all-inclusive calendar. A value of "5C" in the day-of-month field means "the first day included by the calendar on or after the 5th". A value of "1C" in the day-of-week field means "the first day included by the calendar on or after Sunday".

Below are some full examples in Cron-Expression form:

Expression	Meaning

0 0 12 * * ?	Fire at 12pm (noon) every day
0 44 14 ? 3 WED	Fire at 2:44pm every Wednesday in the month of March
0 15 10 15 * ?	Fire at 10:15am on the 15th day of every month
0 15 10 L * ?	Fire at 10:15am on the last day of every month
0 15 10 ? * 6L	Fire at 10:15am on the last Friday of every month
0 15 10 ? * 6#3	Fire at 10:15am on the third Friday of every month
Pay attention to the effects of '?' and '' in the day-of-week and day-of-month fields.*	

If you'd like to know more about the scheduling framework and about the usage of the Cron expressions in the Execution tag, please refer to http://www.opensymphony.com/quartz/wikidocs/TutorialLesson6.html

1.18 Account Group

The Account group holds all information regarding a credit card, wallet or bank account. Many tags depend on the chosen payment method.

The Account tag for a Credit Card looks as follows:

```
<Account>
   <Holder>Joe Doe</Holder>
   <Number>1234 1234 1234 1234</Number>
   <Brand>VISA</Brand>
   <Expiry month="09" year="2005"></Expiry>
   <Verification>123</Verification>
</Account>
```

The Account tag for SEPA looks as follows:

```
<Account>
   <Holder>Joe Doe</Holder>
   <Number>DE23100000001234567890</Number>
   <Bank>MARKDEF1100</Bank>
  <Country>DE</Country>
</Account>
```

The Account tag for User Account (some Virtual Account providers) looks as follows:

```
<Account>
   <Id>joe.doe@mailserver.net</Id>
   <Brand>TESTPAY_WALLET</Brand>
   <Password>1234abcd</Password>
</Account>
```

For a complete list of all possible values and more specific length restrictions, see the chapters about the different payment methods later in this document. Below is a quick overview:

Tags of Account	Data	Length	Mandatory	Description

	Type		/ Optional	
Holder	Alpha	4..128	Cond. Mandatory	Holder of the credit card or bank account. If it can be assumed that shopper is the holder of the account this field can be concatenated from the given and family name of the Name tags.
Number	Alpha-numeric	3..64	Cond. Mandatory	Number of the credit card or bank account. Includes also possible check digits.
Id	Alpha-numeric	3..128	Cond. Mandatory	Typically an email-address uniquely identifying an User Account.
Password	Alpha-numeric	8...32	Cond. Mandatory	The password for the Id. Only used for User Accounts.
Brand	Alpha	3..10	Cond. Mandatory	Name of the credit or debit card brand. The complete list of supported brands is available for download here: https://test.ctpe.io/payment/codes/ccBrands.jsp
Expiry/@month	Numeric	2	Cond. Mandatory	Expiration month of the card in use.
Expiry/@year	Numeric	4	Cond. Mandatory	Expiration year of the card in use.
Start/@month	Numeric	2	Cond. Mandatory	Start month of the used card, mandatory for Switch/Solo if printed on card
Start/@year	Numeric	4	Cond. Mandatory	Start year of the used card, mandatory for Switch/Solo if printed on card
CardIssueNumber	Numeric	1..2	Cond. Mandatory	Number of the issued card. Typically this is mandatory for Switch/Solo or MAESTRO processing and is printed on the front of these cards.
Verification	Numeric	3..4	Cond. Mandatory	The verification number of the credit card (CVV2, CVC2, FDBC, etc.).
Bank	Alpha-numeric	0..12	Cond. Mandatory	The domestic code of the bank to be debited.
BankName	Alpha-numeric	0..50	Cond. Mandatory	The bank name of the bank. Mandatory for Online Bank Transfer (OT) in Austria (EPS) and Netherlands (NL). See document "Asynchronous Workflows" for further details.
Country	Alpha	2 or 4	Cond. Mandatory	Country code according to the ISO 3166-1 specification. For SEPA DD country code "SEPA" is allowed. It can be limited to specific countries (DE, IT, ES, etc.), if SEPA-wide payments shall not be accepted.
Limit	#0.00	1..10,2	Cond. Mandatory	Maximum preauthorization (PA), debit (DB) or credit (CD) amount for a single transaction on a specific account. Generally this tag is optional, yet certain direct debit schemes require the declaration of a limit (e.g. Norway).
Identification	Alpha-numeric	1..16	Cond. Mandatory	Certain direct debit processes require a custom identification number. Currently used e.g. for Norway, Belgium and Italy.

Balance	#0.00	1..10,2	Optional	The Balance of a Virtual Account. Only set in response messages to queries.
RegistrationUrl	Alpha-numeric	10..256	Cond. Mandatory	Especially used for paper based direct debit schemes in order to give the shopper a URL where he or she can download a corresponding paper form, which then is completed and signed. This parameter can only be found in response messages.

1.19 Customer Group

The customer group contains all customer specific information like name, address, and contact details. The name and address tags are used basically for risk management purposes, while the contact tag is important call back validations and transmission of the templates for direct debit countries which require a written confirmation.

The customer tag is mandatory. If customer data are not available then send in dummy data can be sent as a substitute. Doing that – of course – means that risk management will not be usable.

The details tag contains very specific customer information that is necessary for some advanced risk management checks or dedicated direct debit schemes (e.g. "domiciliation bancaria" in Spain).

```
<Customer>
   <Name>
      <Given>Joe</Given>
      <Family>Doe</Family>
       <Birthdate>1970-06-13</Birthdate>
        <Sex>F</Sex>
   </Name>
   <Address>
      <Street>Leopoldstr. 1</Street>
      <Zip>80798</Zip>
      <City>München</City>
      <State>BY</State>
      <Country>DE</Country>
   </Address>
   <Contact>
      <Email>info@provider.com</Email>
      <Ip>123.123.123.123</Ip>
   </Contact>
   <Details>
      <Identity paper="IDCARD">1234567890</Identity>
   </Details>
       </Customer>
```

The tags mentioned above can be populated with the following data:

Name Tag	Data Type	Length	Mandatory / Optional	Description
Salutation	Alpha	1..20	Optional	Salutation of the shopper. The complete list of supported salutations is available for download here: https://test.ctpe.io/payment/codes/salutation.jsp
Title	Alphanumeric	1..20	Optional	Title of the shopper. The complete list of supported titles is available for download here: https://test.ctpe.io/payment/codes/title.jsp
Given	Alpha	2..50	Mandatory	Given (first) name of the shopper.
Family	Alpha	2..50	Mandatory	Family name of the shopper.
Sex	Alpha	1	Optional	Sex of the shopper, 'M' for male or 'F' for female
Birthdate	Alphanumeric	10	Optional	Date in the format yyyy-MM-dd, e.g. 1970-09-12
Company	Alphanumeric	2..40	Optional	Company name of the shopper.

Address Tag	Data Type	Length	Mandatory / Optional	Description
Street	Alphanumeric	5..50	Mandatory	Street and street number of the shopper.
Zip	Alphanumeric	1..10	Mandatory	ZIP code of the shopper's residence.
City	Alpha	2..30	Mandatory	City or location where the shopper lives.
State	Alpha	2..8	Optional	State of the shopper's residence. Not required for many countries. The complete list of supported states is available for download here: https://test.ctpe.io/payment/codes/states.jsp
Country	Alpha	2	Mandatory	Country code according to the ISO 3166-1 specification. The complete list of supported countries is available for download here: https://test.ctpe.io/payment/codes/country.jsp

Contact Tag	Data Type	Length	Mandatory / Optional	Description
Phone	Alphanumeric	8..64	Optional	Used for risk management purposes. Must start with a digit or a '+', at least 7 and max 25 chars long
Mobile	Alphanumeric	10..64	Optional	Used for risk management purposes. Must start with a digit or a '+', at least 7 and max 25 chars long
Email	Alphanumeric	6..128	Mandatory	Used for risk management purposes and transmission of direct debit requests.

Ip	IPv4 or IPv6 format	IPv4: 7 to 15 digits IPv6: 2 to 39 digits	Mandatory	IP number of shopper. Used for risk management purposes. Both IPv4 and IPv6 formats are validated by the customer validation. Currently only IPv4 IPs are accepted on IP white and black-listing. Data type examples: IPv4: 0.0.0.0 / 201.100.133.001 IPv6: 2001:0db8:0000:0000:0000:ff00:0042:8329

Element of Details	Data Type	Length	Mandatory / Optional	Description
Identity/@paper	IDCARD PASSPORT TAXSTATEMENT	5..12	Cond. Mandatory	Type of identity paper
Identity	Alphanumeric	8..64	Cond. Mandatory	Number of identity paper

1.20 MarketData Group

Merchants who want to process transactions containing flight ticket information can do this by using the Tag MarketData in settlement requests, i.e. in requests with transaction type DB or CP.
The data that are actually mandatory to fill in for a valid request depend on the payment processor receiving the transaction. Refer to your account manager for the relevant data for your payment processors.
In order to process MarketData on a merchant account, the corresponding type must be configured. E.g. select "Airline" in the dropdown to process airline transactions.

As described below, most of the fields have to be filled with standardized IATA-codes. The system will only do a plausibility-check on these fields, so please remember to check the receiver's settlement responses. Note that some settlement responses may be flat files as opposed to data transmitted directly by our system.

MarketData Tag	Data Type	Length	Mandatory / Optional	Description
@Type	Enumeration, see below	0..255	Cond. Mandatory	Type of the MarketData
PassengerData			Cond. Mandatory	Contains a list of Passengers

Passenger Tag	Data Type	Length	Mandatory / Optional	Description
@restrictedTicket	Alphanumeric	0..5	Cond.	One of: "true", "false"

			Mandatory	
PassengerName	Alphanumeric	0..255	Mandatory	Name of the Passenger
OriginatingAirport	Alphanumeric	3	Mandatory	IATA Code of the originating Airport
TicketNumber	Alphanumeric	0..32	Cond. Mandatory	The Ticketnumber
AirlineCode	Alphanumeric	0..15	Cond. Mandatory	IATA Airline code
AirlineName	Alphanumeric	0..255	Cond. Mandatory	Name of the Airline
CheckDigit	Numeric	2	Cond. Mandatory	Airline ticket check digit. Only for Diners
AgentCode	Numeric	8	Cond. Mandatory	IATA Agent code
AgentName	Alphanumeric	0..255	Cond. Mandatory	Agent name
Itinerary			Cond. Mandatory	Contains a list of FlightLegs

FlightLeg Tag	Data Type	Length	Mandatory / Optional	Description
@number	Numeric	1..2	Mandatory	Counting the number of legs. Must be in the correct order starting from 1…n legs if there is more than one leg.
@stopOverAllowed	Alphanumeric	0..5	Cond. Mandatory	One of: "true", "false"
CarrierCode	Alphanumeric	0..4	Mandatory	IATA-Code that identifies a carrier on arrival/departure signs, baggage tags, Global Distribution System (GDS), tickets, etc
Destination	Alphanumeric	3	Mandatory	IATA Code of the Destination Airport
FlightNumber	Alphanumeric	0..7	Cond. Mandatory	Normally 3-5 digits + 2 digits airline code.
ClassOfService	Alphanumeric	0..3	Cond. Mandatory	(e.g F for First Class, B for Business,...)
LegDepartureDate	Timestamp	2	Cond. Mandatory	Departure date must be in the Format: YYYY-MM-DD
FareBasis	Alphanumeric	0..6	Cond.	Code for Airline charges

			Mandatory	

Sample Data:

```
<MarketData type="Airline">
  <PassengerData>
    <Passenger restrictedTicket="true">
      <PassengerName>Ettone Bewarzer</PassengerName>
      <OriginatingAirport>BEG</OriginatingAirport>
      <AirlineCode>JAT</AirlineCode>
      <Itinerary>
        <FlightLeg number="1" stopOverAllowed="true">
          <CarrierCode>JT</CarrierCode>
          <Destination>STR</Destination>
          <FlightNumber>JU347</FlightNumber>
          <ClassOfService>F</ClassOfService>
          <LegDepartureDate>2012-12-12</LegDepartureDate>
        </FlightLeg>
      </Itinerary>
    </Passenger>
  </PassengerData>
</MarketData>
```

1.21 Authentication Group

This tag group allows the merchant to submit any kind of Authentication information that authenticates the transaction itself (such as 3D-Secure, SMS-identification and the like).

Currently this tag group is used to submit authentication information gathered from a Verified by Visa (VbV) or Mastercard Secure Code (MSC) request performed by a merchant himself.

See the document "Asynchronous Workflow" for more details about 3D-Secure processes. Further information is also available in the "ThreeDSecure_Integration" document.

```
<Authentication type="3DSecure">
       <Parameter name="VERIFICATION_ID">AAACAgSRBklmQCFgMpEGAAAAAAA=</Parameter>
       <Parameter name="XID">CAACCVVUlwCXUyhQNlSXAAAAAAA=</Parameter>
       <ResultIndicator>05</ResultIndicator>
</Authentication>
```

The Authentication group has one value to determine the type of Authentication.

Values for type	Description
3DSecure	Use this value for the type if the authentication was a 3DSecure process.

Tags of Authentication	Data Type	Length	Mandatory / Optional	Description
Parameter	Alphanumeric	1..128	Conditional	Contains the name and value of a parameter of the specified Authentication type
ResultIndicator	Alphanumeric	1..128	Optional	Contains the result of the Authentication process. For 3D-Secure this must be one of the following (ECI-Value): 01 = MASTER_3D_ATTEMPT 02 = MASTER_3D_SUCCESS 05 = VISA_3D_SUCCESS 06 = VISA_3D_ATTEMPT 07 = DEFAULT_E_COMMERCE

Values for Parameter name	Description
VERIFICATION_ID	Verification ID for the 3D-Secure authentication. This is the CAVV value for VISA or the UCAF value for Mastercard. Must be Base64 encoded.
XID	XID for Verified By Visa (VbV) or Mastercard Secure Code (MSC) transactions. Must be Base64 encoded.

1.22 Frontend Group

The Frontend group is used for three scenarios:

- for merchants that are (potentially) using any kind of asynchronous workflow like Online Bank Transfer, Verified By VISA or Mastercard Securecode.
- to specify a Response URL to be notified if the state of a Registration has been changed.
- For any asynchronous workflow like Wallet payments (Moneybookers, iDeal, EPS, Interac, …)

This tag group is mandatory in the XML request message if using an asynchronous payment method..
Please refer to the document "Asynchronous Workflow" for an explanation how these tags are utilized.

```
<Frontend>
   < ResponseUrl >https://myshop.com/payment/result.php</ResponseUrl>
      <SessionID>89asdg89sdg98sfksdf8907</SessionID>
</Frontend>
```

Tag of Analysis	Data Type	Length	Mandatory / Optional	Description
ResponseUrl	Alphanumeric, URL	255	Mandatory	The response URL of the merchant where the asynchronously generated result of the

				transaction is posted to. The ResponseUrl MUST use either port 443 (https) or port 80 (http). Other ports are not supported and will not work! In case of a Registration (RG) or Reregistration (RR) the response XML of the Confirmation (CF) gets sent to this URL only if the RG was not confirmed immediately (refer to <ConfirmationStatus> in the tag group ProcessingResult).
SessionID	Alphanumeric	255	Mandatory	The session ID of the merchant's shop. This session ID gets returned to the merchant as part of the asynchronously generated result of the transaction at the end of the workflow (asynchronous xml response). Can be useful at the end of the payment process to reload the correct session for the end user.

1.23 Analysis Group

The Analysis group allows adding additional customized attributes that are not available by the standard set of XML tags to the transaction.
With the help of these freely definable 'Criteria', a customised statistics and analysis can be performed against all of the transactions which have these parameters included. Furthermore the Criterion can be used to store all kinds of information within the transaction for various purposes such as an automated or manual export and further processing of the information in third party systems. The Criterion group has a purely statistical and informative value.

Possible examples are the turnover of different affiliates, age groups, or any other imaginable value or subdivision that should be available in the analysis front end later.

```
<Analysis>
   <Criterion name="affiliate">ExternalShopXY</Criterion>
   <Criterion name="age">30-40</Criterion>
   <Criterion name="known">yes</Criterion>
   <Criterion name="customerid">1234567</Criterion>
  </Analysis>
```

An example for the storage of information within the transaction would be to submit (for example) an external customer ID or a special token for external matching purposes as Criterion.

Tag of Analysis	Data Type	Length	Mandatory / Optional	Description
Criterion	Alphanumeric	0..1024	Optional	Freely definable value for a specific criterion.

Value for name	Description
Alphanumeric 1..32	Freely definable name for a specific criterion.

Note: If you add Analysis information to a Schedule (SD) Transaction, the fired Transactions also contain this Analysis tag group.

Criterions are also used for special features of payment processors like iDeal, Sofortueberweisung or Moneybookers. See the document "Asynchronous Workflows" for processor-specific details.

1.24 Processing Group

The processing group contains a summary of the processing of a transaction. The structure of the status and reason codes is hierarchical while the return code is an independent, internal value that is used for very specific return messages. Any merchant-side matching should be performed on the processing code or status and reason codes only, rather than on the respective messages associated to these statuses.

```
<Processing code="DD.DB.90.00">
   <Timestamp>2003-02-12 14:58:07</Timestamp>
   <Result>ACK</Result>
   <Status code="90">NEW</Status>
   <Reason code="00">Successful Processing</Reason>
   <Return code="000.000.000">Transaction succeeded</Return>
   <Risk score="-20"></Risk>
</Processing>
```

The Result tag in the processing group derives from the status of the transaction, which means that a transaction that has the status REJECTED or FAILED has the result NOK, while all other statuses result in an ACK. For each status there are one or several reasons.

Please refer to the appendices for a complete list of all status and reason codes. All current return codes are listed at https://test.ctpe.io/payment/codes/mainMenu.jsp

Tag of Processing	Data Type	Length	Mandatory / Optional	Description
Timestamp	Timestamp	19	Mandatory	Date and time when the transaction was processed.
Result	ACK, NOK	3	Mandatory	In the case of the status REJECTED or FAILED the result is NOK (Not OK). In all other cases the result is ACK (Acknowledgement).
Status	Alphanumeric	0..19	Mandatory	Status message which belongs to the Status code (e.g. NEW, REJECTED …). Important: Please use the status code and not the associated reason-

				or return message for matching purposes.
Reason	Alphanumeric	0..64	Mandatory	Reason message that belongs to the Reason code. (e.g. Successful Processing, Account Validation, Bank Code Validation …). Important: Please use the status code and not the associated reason- or return message for matching purposes.
Return	Alphanumeric	0..256	Mandatory	Return message that belongs to the Return code (e.g. Validation Algorithm DE102 failed, …). Important: Please use the status code and not the associated reason- or return message for matching purposes.
ConfirmationStatus	CONFIRMED, PENDING	0..20	Optional	In case of a response to a Registration (RG) or Reregistration (RR) request, this status message tells if the registration was auto-confirmed immediately (CONFIRMED) or is waiting (PENDING) for a confirmation. A debit request (DB) can only be submitted if the Registration was confirmed.
InfoMessage	Alphanumeric	0..256	Optional	Additional Information for the integrator. Can only occur on the test system while testing integration. May contain information that the returned error code was forced for testing purposes. More about forcing error codes in is available in the "Technical Quick Start" document.
Risk/@score	Numeric		Optional	Contains the risk score for the executed transaction (for a payment transaction the score of the corresponding RM.RI transaction). This value is only returned if risk operations (e.g. blacklist, velocity checks …) were executed.

Values for Processing code	Description
AA.AA.00.00	The processing code of a transaction. It's a simple concatenation of method, type, status and reason code. It provides the context within a status or reason has appeared and contains together

	with the return code all information about the processing of a transaction.
Values for Status code	Description
00	The status code of a transaction. See https://test.ctpe.io/payment/codes/statusCodes.jsp for a complete list.
Values for Reason code	Description
00	The reason code of a transaction. Every status has one or several reasons. See https://test.ctpe.io/payment/codes/reasonCodes.jsp for a complete list of status and reason code combinations.
Values for Return code	Description
000.000.000	The return ID of a transaction. See https://test.ctpe.io/payment/codes/resultCodes.jsp for a list of the current codes.

1.24.1 Asynchronous Response Processing Group

For asynchronous response messages an additional subgroup called "Redirect" is part of the response message.

```
<Transaction … response="ASYNC" …>
…
<Processing code="CC.DB.80.00">
       <Timestamp>2003-02-12 14:58:07</Timestamp>
       <Result>ACK</Result>
       <Status code="80">WAITING</Status>
       <Reason code="00">User enrolled, awaiting 3D ...</Reason>
       <Return code="000.200.000">Transaction succeeded. ...</Return>
       <Redirect url="https://www.mybank.com/3D_validation">
              <Parameter
name="TermUrl">https://test.ctpe.io/payment/3D_response</Parameter>
              <Parameter
name="PAReq">m123n456o789p876q543r22323145346576</Parameter>
          <Parameter name="MD">24358432975908324758904327589434</Parameter>
     </ Redirect>
</Processing>
```

Depending on the type of the asynchronous process (i.e. Online Bank Transfer, Verified By VISA, Mastercard Securecode, etc.) the Redirect subgroup contains a number of different tags.
Typically the merchant redirects the end user's browser to the Redirect URL and passes the other parameters in this subgroup as parameters to the Redirect URL.
Please refer to the "Asynchronous Workflow" document for an explanation of how these tags must be processed by the merchant to fulfill the workflow requirements of the asynchronous process.

Attributes and Tags of Redirect	Data Type	Length	Mandatory / Optional	Description

URL	Alphanumeric, URL	2048	Mandatory	URL that must be called by the merchant in order to proceed. The merchant redirects the browser to this URL.
Parameter	Alphanumeric	4096	Optional	Any kind of parameter needed for the workflow. Each Parameter needs to be posted to the URL.

1.25 Connector Group

Within the Connector group information about the connector that was selected for processing of the transaction is given back. This is always used in conjunction with Prepayment (PP) or Invoice (IV) transactions where the shopper needs to know to which bank account a payment must be directed. Furthermore it is necessary to identify the receiving bank account in the context of direct debit registrations (DD.RG).

```
<Connector>
   <Account>
      <Holder>Internal Account Holder</Holder>
      <Number>DE82700202700666898869</Number>
      <Bank>HYVEDEMM</Bank>
      <Country>DE</Country>
   </Account>
</Connector>
```

Tags of Account	Data Type	Length	Mandatory / Optional	Description
Holder	Alpha	4..128	Mandatory	Holder of the bank account.
Number	Alphanumeric	3..64	Cond. Mand.	International Bank Account Number of the shopper's processing bank account. This number can be used also by foreign customers to make cross-border credit transfers. Also includes possible check digits.
Bank	Alphanumeric	0..12	Cond. Mand.	Bank Identifier Code (BIC/SWIFT) of the bank that holds the direct debit or credit transfer account. This bank code can be used also by foreign customers to make cross-border credit transfers.
Country	Alpha	2	Cond. Mand.	Country code according to the ISO 3166-1 specification.

1.26 Relevance Group

The Relevance group is an option to describe in which relevance a transaction is being processed. Typically this option is used in the context of User Accounts or Virtual Accounts only.

A typical example for that looks like this:

```
<Relevances>
    <Relevance>DEFAULT_PAY</Relevance>
</Relevances>
```

Values of Relevance	Description
DEFAULT_DEBIT	Marks a debit registration as the default payment method for the User Account.
DEFAULT_CREDIT	Marks a credit registration as the default transfer payment method for the User Account.
DEFAULT_UNLOAD	Marks a payment registration as the default account to cash the Virtual Account out to.
DEFAULT_LOAD	Marks a payment registration as the default account where the Virtual Account should be (re)charged from.
SOURCE	Marks a payment transaction (debit) as a transaction that was used to (re)charge a Virtual Account from.
TARGET	Marks a payment transaction (credit) as a transaction that was used to cash a Virtual Account out to.

2 Direct Debit (DD) Transactions

For the payment method Direct Debit (DD) the following transaction types can be used:

- Debit (DB)
- Reversal (RV)
- Refund (RF)

When a direct debit is returned either by the bank (rejection) or by the shopper (revocation) a chargeback is initiated and gets delivered back as a response xml file.

- Chargeback (CB)

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

2.1 Debit (DB)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ShopperID>shopper00001234</ShopperID>
      </Identification>
      <Payment code="DD.DB">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
```

```
      </Payment>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
         <Country>DE</Country>
      </Account>
      <Customer>
         <Name>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ShopperID>shopper00001234</ShopperID>
      </Identification>
      <Processing code="DD.DB.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
```

```
   </Transaction>
</Response>
```

2.2 SEPA Debit (DB)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="DD.DB">
         <Presentation>
            <DueDate>2013-08-13</DueDate>
            <Mandate id="arbitrary-mandate-identifier" dateOfSignature="2013-08-01"/>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Recurrence mode="INITIAL"/>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
        <Country>DE</Country>
      </Account>
      <Customer>
          <Name>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k092l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DB">
            <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2013-08-13</Date>
            <DueDate>2013-08-13</DueDate>
            <Mandate id="arbitrary-mandate-identifier" dateOfSignature="2013-08-01"/>
            <DueDate>2013-08-13</DueDate>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

2.3 Reversals (RV)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="DD.RV"/>
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="DD.RV.00.05">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Transaction Reversed</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.RV">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
  </Transaction>
</Response>
```

2.4 Refunds (RF)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="DD.RF">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
```

```
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="DD.RF.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.RF">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

2.5 Chargebacks (CB)

Chargeback transactions are not triggered by the merchant, but retrieved from the bank by the system using a scheduled job. The daily CSV file with all Chargeback transactions can be downloaded from the Business Intelligence Platform or queried with XML through the query API. *Note: not all payment processors support chargeback reporting through the system. Please verify your payment processor's ability to do such with your account manager.*

Please note that a chargeback transaction is a new transaction of its own with a new Unique ID. The Reference ID refers to the Unique ID of the transaction that was returned. The Presentation amount is the amount returned by the bank and not necessarily the same as the presentation amount of the referencing transaction (this is especially important if a currency conversion took place).

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="ASYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="DD.CB.00.10">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="10">Reason not Specified</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.CB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>1234.1234.1234 - Order Number 1234</Descriptor>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

3 Credit Transfer (CT) Transactions

A credit transfer (CT) transaction is a standard bank transfer transaction, which credits the bank account of the shopper and debits the merchant account.

Credit Transfer Transaction Types

- Credit (CD)
- Reversal (RV)
- Receipt (RC)

Introductory remark:
When using the examples below please, replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

3.1 Credit (CD)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="CT.CD">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
         <Country>DE</Country>
```

```
      </Account>
      <Customer>
         <Name>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
      </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k0981765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="CT.CD.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CT.CD">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

3.2 Reversal (RV)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="CT.RV"/>
    </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CT.RV.00.05">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="05">Transaction Reversed</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CT.RV">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

3.3 Receipt (RC)

Receipt transactions are not triggered by the merchant, but are retrieved from the bank by the system using a scheduled job. The daily XML file with all Receipt transactions can be fetched via the XML Query interface or downloaded from the Business Intelligence Platform (BIP).

Please note that a Receipt transaction is a new transaction of its own with a new Unique ID. The Reference ID refers to the Unique ID of the transaction that was returned. The Presentation amount is the amount given back by the bank and not necessarily the same as the presentation amount of the referencing transaction (this is especially important if a currency conversion took place).

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="ASYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CT.RC.00.10">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="10">Reason not Specified</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CT.RC">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>1234.1234.1234 - Order Number 1234</Descriptor>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

4 Prepayment (PP) and Invoice (IV) Transactions

Prepayment (PP) and Invoice (IV) transactions both follow an identical payment process. In both cases the shopper triggers a credit transfer to the account of the merchant. The only difference is that the service / goods is delivered before (Invoice) or after (Prepayment) the receipt of the money.

Upon receipt of the money, every incoming shopper credit transfer generates a Receipt (RC) transaction that contains a reference to the prior Preauthorisation or Registration. This receipt gets subsequently matched with the respective Prepayment or Invoice transaction.

If a received prepayment amount should be returned to the customer, a Refund (RF) transaction can be triggered, so there are overall 3 supported payment types for Prepayment / Invoice:

- Preauthorisation (PA)
- Receipt (RC)
- Refund (RF)

All following examples are done for Prepayment transactions. In case of Invoice transactions only 'PP' inside the payment code has to be replaced by 'IV'.

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

4.1 Preauthorisation (PA)

In general a Preauthorisation (PA) is always used to preauthorise a credit card transaction. This means that all Risk Management checks are performed and additionally the credit card issuer validates the payment and (in case of a credit card payment) the amount of the transaction gets "reserved" and deducted from the shopper's credit limit. For all Preauthorised invoice transactions the Descriptor tag gets returned which contains the exact text the shopper should quote on his credit transfer.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="xxx" />
```

```
   </Header>
   <Transaction mode="LIVE" channel="xx" response="SYNC">
      <Identification>
         <TransactionID>Merchant Assigned ID</TransactionID>
      </Identification>
      <User login="xx" pwd="xx"/>
      <Payment code="PP.PA">
         <Memo>Please send me more infos</Memo>
         <Presentation>
            <Amount>11.11</Amount>
            <Currency>EUR</Currency>
            <Usage/>
         </Presentation>
      </Payment>
      <Customer>
         <Name>
            <Family>Portisch</Family>
            <Given>Hugo</Given>
            <Salutation/>
         </Name>
         <Contact>
            <Email>bob@kosel.com</Email>
            <Ip>178.158.237.155</Ip>
            <Mobile>+49 98 520 2990</Mobile>
            <Phone>+49 179 520 2990</Phone>
         </Contact>
         <Address>
            <City>Wien</City>
            <Country>AT</Country>
            <State>AT13</State>
            <Street>Stephansplatz 2</Street>
            <Zip>1011</Zip>
         </Address>
      </Customer>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" channel="xxx" response="SYNC">
      <Identification>
         <ShortID>8255.8459.7000</ShortID>
         <UniqueID>d225a9fe00be7daa0100bfeb89f20b3e</UniqueID>
         <TransactionID>3146</TransactionID>
         <ReferenceID/>
      </Identification>
      <Payment code="PP.PA">
         <Clearing>
            <Amount>11.11</Amount>
            <Currency>EUR</Currency>
            <Descriptor>8255.8459.7000</Descriptor>
            <FxDate>2004-12-11 02:18:32</FxDate>
            <FxRate>1.0</FxRate>
            <FxSource>INTERN</FxSource>
         </Clearing>
```

```
      </Payment>
      <Processing code="90.00">
         <Timestamp>2004-12-11 02:18:32</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.00">Request successfully processed in 'Merchant in
Integrator Test Mode'</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Connector>
         <Account>
            <Number>NL89INGB0672920301</Number>
            <Holder>Demomerchant GmbH</Holder>
            <Bank>INGBNL2A</Bank>
            <Country>NL</Country>
         </Account>
      </Connector>
   </Transaction>
</Response>
```

4.2 Receipt (RC)

A receipt is an incoming credit transfer on the merchant account. If the receipt gets matched with a prior Preauthorisation transaction, the Reference ID of the receipt is the Unique ID of this Preauthorisation. Another option is to match the transaction with a permanent expected descriptor provided within a prior Registration transaction. In this case the Reference ID contains the Unique ID of the Registration.

A Receipt is not triggered by the merchant, but can be fetched either through the export menu inside the BIP or by using the XML Integrator (Queries) specification.

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="ASYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="00.80">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="80">Matched Consumer Credit Transfer</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="PP.RC">
         <Clearing>
            <Amount>1.00</Amount>
```

```
            <Currency>EUR</Currency>
            <Descriptor>1234.1234.1234 - Order Number 1234</Descriptor>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

4.3 Refund (RF)

Every Receipt can be refunded by providing the Unique ID of the Receipt as Reference ID.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="PP.RF">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="PP.RF">
         <Clearing>
            <Amount>1.00</Amount>
```

```
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

5 Credit Card (CC) Transactions

For Credit Card transactions the following payment types are supported:

- Debit
- Preauthorisation (and Reauthorization)
- Capture
- Reversal
- Refund
- Chargeback

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

5.1 Debit (DB)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="CC.DB">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Recurrence mode="INITIAL"/>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>1234123412341234</Number>
         <Brand>VISA</Brand>
```

```
         <Expiry month="09" year="2009"></Expiry>
         <Verification>123</Verification>
      </Account>
      <Customer>
         <Name>
            <Salutation>MR</Salutation>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Phone>+49-89-1234566</Phone>
            <Mobile>+49-172-1234566</Mobile>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
      <Analysis>
         <Criterion name="affiliate">ExternalShopXY</Criterion>
         <Criterion name="age">30-40</Criterion>
         <Criterion name="known">yes</Criterion>
         <Criterion name="customerid">1234567</Criterion>
      </Analysis>
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="CC.DB.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.DB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
```

```
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

5.2 Preauthorizations (PA)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="CC.PA">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>1234123412341234</Number>
         <Brand>VISA</Brand>
         <Expiry month="09" year="2005‟></Expiry>
         <Verification>123</Verification>
      </Account>
      <Customer>
         <Name>
            <Salutation>MR</Salutation>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Phone>+49-89-1234566</Phone>
            <Mobile>+49-172-1234566</Mobile>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
```

```
      </Customer>
      <Analysis>
         <Criterion name="affiliate">ExternalShopXY</Criterion>
         <Criterion name="age">30-40</Criterion>
         <Criterion name="customerid">1234567</Criterion>
      </Analysis>
</Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="CC.PA.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.PA">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
  </Transaction>
</Response>
```

5.3 Reauthorization (PA)

The Reauthorization allows for the updating of an existing preauthorization.
Please check with your account manager if the configured Credit Card acquirer supports this functionality or not!

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
```

```
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
      </Identification>
      <Payment code="CC.PA">
         <Presentation>
            <Amount>2.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
    </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="CC.PA.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.PA">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

5.4 Captures (CP)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="CC.CP">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Analysis>
         <Criterion name="affiliate">ExternalShopXY</Criterion>
         <Criterion name="age">30-40</Criterion>
         <Criterion name="known">yes</Criterion>
         <Criterion name="customerid">1234567</Criterion>
      </Analysis>
    </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CC.CP.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.CP">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de  1234.1234.1234  +49  (89)  12345  678  Order  Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
```

```
  </Transaction>
</Request>
```

5.5 Reversals (RV)

Please check with your account manager if the configured Credit Card acquirer supports this functionality or not!

The Reversal voids a previous Debit (DB) or Preauthorization (PA) transaction.

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="CC.RV"/>
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CC.RV.00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="05">Transaction Reversed</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.RV">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
```

```
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Request>
```

5.6 Refunds (RF)

Refund Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="CC.RF">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
     </Transaction>
```

Refund Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CC.RF.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.RF">
         <Clearing>
```

```
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
    </Transaction>
</Request>
```

5.7 Chargebacks (CB)

Chargeback transactions are not triggered by the merchant but retrieved from the bank by the system using a scheduled job. The daily XML file with all Chargeback transactions can be fetched from a predefined SFTP directory or downloaded from the Business Intelligence Platform.

Please note that a chargeback transaction is a new transaction of its own with a new Unique ID. The Reference ID refers to the Unique ID of the transaction which was returned. The Presentation amount is the amount given back by the bank and not necessarily the same as the presentation amount of the referencing transaction (this is especially important if a currency conversion took place).

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="ASYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="CC.CB.00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="40">Revocation or Dispute</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="CC.CB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
```

```
   </Transaction>
</Response>
```

6 Virtual Account (VA) Transactions for Klarna Invoice

With the Klarna Invoice workflow, the shopper receives an invoice for the total amount and can pay after having received the products.

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

The following payment types can be executed:

Payment type	Description
PA	Preauthorization of the amount
PA referencing other PA	Change the preauthorized amount
CP	Capture of the preauthorized amount
Partial CP	Partial capture of the preauthorized amount
RV	Reversal of the transaction
RF	Refund of the captured transaction
Partial RF	Partial refund of the captured transaction

6.1 Klarna Invoice Payment – Preauthorization (PA)

The following rules apply for the integration with Klarna Invoice:
- Brand to be used is "KLARNA_INVOICE"
- The payment method is Virtual Account "VA"
- Payment type to initiate a Klarna Invoice transaction is "PA"

The complete Payment tag looks looks like:

```
<Payment code="VA.PA">
```

The account tag looks like:

```
<Account>
      <Brand>KLARNA_INVOICE<Brand>
</Account>
```

Initial Payment Request

If the provided channel is configured for KLARNA_INVOICE, the XML payment request is formatted as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" channel="xx" source="XML" response="SYNC">
    <User login="xx" pwd="xx"/>
    <Identification>
      <TransactionID>MerchantAssignedID</TransactionID>
    </Identification>
    <Payment code="VA.PA">
      <Presentation>
        <Amount>12.95</Amount>
        <Currency>EUR</Currency>
        <Usage>Order Number 1234</Usage>
      </Presentation>
    </Payment>
    <Account>
      <Brand>KLARNA_INVOICE</Brand>
    </Account>
    <Customer>
            <Name>
                <Family>Approved</Family>
                <Given>Testperson-de</Given>
                <Sex>M</Sex>
                <Birthdate>1960-07-07</Birthdate>
            </Name>
            <Contact>
                <Ip>127.0.0.1</Ip>
                <Email>test-email@your-domain.com</Email>
                <Phone>01522113356</Phone>
            </Contact>
            <Address>
                <Street>Hellersbergstraße 14</Street>
                <City>Neuss</City>
                <Zip>41460</Zip>
                <Country>DE</Country>
            </Address>
        </Customer>
        <Analysis>
           <Criterion name="KLARNA_CART_ITEMSTOTAL">4</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_ARTICLE_NR">1</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_TITLE">Product 1</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_UNIT_PRICE">1.00</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_VAT">6.0</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_DISCOUNT">0.0</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_FLAGS">32</Criterion>
           <Criterion name="KLARNA_CART_ITEM1_QUANTITY">5</Criterion>
           <Criterion name="KLARNA_CART_ITEM2_ARTICLE_NR">2</Criterion>
           <Criterion name="KLARNA_CART_ITEM2_TITLE">Product 2</Criterion>
           <Criterion name="KLARNA_CART_ITEM2_UNIT_PRICE">5.00</Criterion>
```

```
            <Criterion name="KLARNA_CART_ITEM2_VAT">6.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_FLAGS">32</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_QUANTITY">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_ARTICLE_NR">3</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_TITLE">Shipping fee</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_UNIT_PRICE">1.95</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_VAT">19.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_FLAGS">40</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_QUANTITY">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_ARTICLE_NR">4</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_TITLE">Invoice fee</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_UNIT_PRICE">1.00</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_VAT">19.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_FLAGS">48</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_QUANTITY">1</Criterion>
         </Analysis>
      </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="x">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>UniqueID</UniqueID>
         <ShortID>ShortID</ShortID>
      </Identification>
      <Processing code="VA.PA.90.00">
         <Timestamp>2013-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <Risk score="0"
      </Processing>
      <Payment code="VA.PA">
         <Clearing>
            <Amount>12.95</Amount>
            <Currency>EUR</Currency>
            <Descriptor>ShortID Channelname Order Number 1234</Descriptor>
            <FxRate>1.0</FxRate>
            <FxSource>INTERN</FxSource>
            <FxDate>2013-07-29 11:33:39</FxDate>
         </Clearing>
      </Payment>
  </Transaction>
</Response>
```

After the PA is sent, Klarna can respond with three outcome scenarios;

Response	Connector response data	Description
Accepted	Risk score (integer 1) and Klarna's reservation number	Klarna has accepted the purchase; please redirect the shopper to a success page.
Pending	Risk score (integer 2) and Klarna's reservation number	Klarna suspects fraud and has marked the transaction for manual review. Please redirect the shopper to a success page. Klarna will review the transaction within a maximum of 24 hours, until that time the products should not be shipped.
Denied	Klarna's error code and error message	Klarna has rejected the transaction, please redirect the shopper to a failure page and display the error message to the shopper.

6.1.1 General XML parameters for Preauthorization (PA)

The following are mandatory when using Klarna:

6.1.1.1 CUSTOMER group parameters (mandatory):

The CUSTOMER group parameters are mandatory to send in and needed by Klarna in order to determine the credit scoring and the risk level of the shopper.

```
<Customer>
      <Name>
          <Family>Approved</Family>
          <Given>Testperson-de</Given>
          <Sex>M</Sex>
          <Birthdate>1960-07-07</Birthdate>
      </Name>
      <Contact>
          <Ip>127.0.0.1</Ip>
          <Email>test-email@your-domain.com</Email>
          <Phone>01522113356</Phone>
      </Contact>
      <Address>
          <Street>Hellersbergstraße 14</Street>
          <City>Neuss</City>
          <Zip>41460</Zip>
          <Country>DE</Country>
      </Address>
```

```
        </Customer>
```

6.1.1.2 Criterion parameters: Shopping card ITEM parameters (Mandatory)

The `<Criterion>` parameters are mandatory as well; otherwise Klarna cannot create the invoice and send it to the shopper.
The item number of the shopping cart items has to increase by 1 for each shopping cart item. The item number of the first shopping cart item (article number, title, unit price, vat, discount, flags and quantity) will be 1. The second shopping cart item will be 2. In the example code listed below, 4 items are added to a sample cart.

XML CRITERION Parameter	Value	Mandatory
KLARNA_CART_ITEMSTOTAL	Shopping cart items count	Yes
KLARNA_CART_ITEM1_ARTICLE_NR	Article number of one shopping cart item	Yes
KLARNA_CART_ITEM1_TITLE	Description of one shopping cart item	Yes
KLARNA_CART_ITEM1_UNIT_PRICE	VAT-inclusive unit price of the shopping cart item	Yes
KLARNA_CART_ITEM1_VAT	VAT percentage applied to the shopping cart item	Yes
KLARNA_CART_ITEM1_DISCOUNT	Discount percentage applied to the shopping cart item	Yes
KLARNA_CART_ITEM1_FLAGS	Identifying variable that determines the type of shopping cart item	Yes
KLARNA_CART_ITEM1_QUANTITY	Quantity of the shopping cart item	Yes

The analysis tag for the mandatory criterions looks like:

```
<Analysis>
            <Criterion name="KLARNA_CART_ITEMSTOTAL">4</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_ARTICLE_NR">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_TITLE">Product 1</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_UNIT_PRICE">1.00</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_VAT">6.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_FLAGS">32</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_QUANTITY">5</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_ARTICLE_NR">2</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_TITLE">Product 2</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_UNIT_PRICE">5.00</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_VAT">6.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_FLAGS">32</Criterion>
            <Criterion name="KLARNA_CART_ITEM2_QUANTITY">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_ARTICLE_NR">3</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_TITLE">Shipping fee</Criterion>
```

```
            <Criterion name="KLARNA_CART_ITEM3_UNIT_PRICE">1.95</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_VAT">19.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_FLAGS">40</Criterion>
            <Criterion name="KLARNA_CART_ITEM3_QUANTITY">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_ARTICLE_NR">4</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_TITLE">Invoice fee</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_UNIT_PRICE">1.00</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_VAT">19.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_DISCOUNT">0.0</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_FLAGS">48</Criterion>
            <Criterion name="KLARNA_CART_ITEM4_QUANTITY">1</Criterion>
         </Analysis>
```

6.2 Captures (CP)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
      <Transaction mode="CONNECTOR_TEST" channel="xx" response="SYNC" source="XML">
        <Identification>
            <TransactionID>MerchantAssignedID</TransactionID>
            <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
        </Identification>
        <User login="xx" pwd="xx" />
        <Payment code="VA.CP">
             <Presentation>
                <Amount>13.95</Amount>
                <Currency>EUR</Currency>
             </Presentation>
        </Payment>
    </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
<Transaction mode="LIVE" channel="xx" response="SYNC">
       <Identification>
             <TransactionID>MerchantAssignedID</TransactionID>
             <UniqueID> </UniqueID>
             <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
             <ShortID>ShortID</ShortID>
       </Identification>
       <Payment code="VA.CP">
            <Clearing>
                <Amount>13.95</Amount>
```

```
                <Currency>EUR</Currency>
                <Descriptor>ShortID Channelname Order Number 1234</Descriptor>
                <FxRate>1.0</FxRate>
                <FxSource>INTERN</FxSource>
                <FxDate>2013-07-29 12:44:01</FxDate>
            </Clearing>
        </Payment>
        <Processing code="VA.CP.90.00">
            <Timestamp>2013-07-29 12:44:01</Timestamp>
            <Result>ACK</Result>
            <Status code="90">NEW</Status>
            <Reason code="00">Successful Processing</Reason>
            <Return code="000.000.000">Transaction succeeded'</Return>
      </Processing>
</Transaction>
</Response>
```

6.2.1 Partial Captures (CP)

When only a part of the initially ordered products is sent, these products need to be specified in the request with the same CRITERIONS as were used in the Preauthorization request. Multiple captures are supported; the Preauthorization remains valid until the full amount is captured.

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
<Transaction mode="LIVE" channel="xx" response="SYNC" source="XML">
      <Identification>
          <TransactionID>MerchantAssignedID</TransactionID>
          <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
      </Identification>
      <User login="xx" pwd="xx" />
      <Payment code="VA.CP">
            <Presentation>
                  <Amount>4.0</Amount>
                  <Currency>EUR</Currency>
            </Presentation>
       </Payment>
       <Analysis>
            <Criterion name="KLARNA_CART_ITEMSTOTAL">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_ARTICLE_NR">1</Criterion>
            <Criterion name="KLARNA_CART_ITEM1_QUANTITY">4</Criterion>
      </Analysis>
</Transaction>
</Response>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
<Transaction mode="LIVE" channel="xx" response="SYNC">
        <Identification>
             <TransactionID>MerchantAssignedID</TransactionID>
             <UniqueID>UniqueID</UniqueID>
             <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
             <ShortID>ShortID</ShortID>
       </Identification>
         <Payment code="VA.CP">
             <Clearing>
                 <Amount>4.00</Amount>
                 <Currency>EUR</Currency>
                 <Descriptor>ShortID Channelname Order Number 1234</Descriptor>
                 <FxRate>1.0</FxRate>
                 <FxSource>INTERN</FxSource>
                 <FxDate>2013-07-29 12:53:49</FxDate>
             </Clearing>
         </Payment>
         <Processing code="VA.CP.90.00">
             <Timestamp>2013-07-29 12:53:50</Timestamp>
             <Result>ACK</Result>
             <Status code="90">NEW</Status>
             <Reason code="00">Successful Processing</Reason>
             <Return code="000.000.000">Transaction succeeded'</Return>
         </Processing>
    </Transaction>
</Response>
```

6.3 Reversals (RV)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
<Transaction mode="LIVE" channel="xx" response="SYNC" source="XML">
       <Identification>
             <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
       </Identification>
        <User login="xx" pwd="xx" />
        <Payment code="VA.RV"> />
</Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
```

```
<Transaction mode="LIVE" channel="xx" response="SYNC">
        <Identification>
             <TransactionID>MerchantAssignedID</TransactionID>
             <UniqueID>UniqueID</UniqueID>
             <ReferenceID>UniqueId of a Preauthorization</ReferenceID>
             <ShortID>ShortID</ShortID>
       </Identification>
        <Payment code="VA.RV">
             <Clearing>
                 <Amount>13.95</Amount>
                 <Currency>EUR</Currency>
                 <Descriptor>ShortID Channelname Order Number 1234</Descriptor>
                 <FxRate>1.0</FxRate>
                 <FxSource>INTERN</FxSource>
                 <FxDate>2013-07-29 13:06:56</FxDate>
             </Clearing>
         </Payment>
         <Processing code="VA.RV.90.00">
             <Timestamp>2013-07-29 13:06:57</Timestamp>
             <Result>ACK</Result>
             <Status code="90">NEW</Status>
             <Reason code="00">Successful Processing</Reason>
             <Return code="000.000.000">Transaction succeeded'</Return>
          </Processing>
    </Transaction>
</Response>
```

6.4 Refunds (RF)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
<Transaction mode="LIVE" channel="xx" response="SYNC" source="XML">
        <Identification>
             <ReferenceID>UniqueId of a Capture</ReferenceID>
        </Identification>
        <User login="xx" pwd="xx" />
        <Payment code="VA.RF">
              <Presentation>
                    <Amount>13.95</Amount>
                    <Currency>EUR</Currency>
              </Presentation>
        </Payment>
</Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
<Transaction mode="LIVE" channel="xx" response="SYNC">
        <Identification>
             <TransactionID>MerchantAssignedID</TransactionID>
             <UniqueID>UniqueID</UniqueID>
             <ReferenceID>UniqueId of a Capture</ReferenceID>
             <ShortID>ShortID</ShortID>
       </Identification>
         <Payment code="VA.RF">
             <Clearing>
                 <Amount>13.95</Amount>
                 <Currency>EUR</Currency>
                 <Descriptor> ShortID Channelname Order Number 1234</Descriptor>
                 <FxRate>1.0</FxRate>
                 <FxSource>INTERN</FxSource>
                 <FxDate>2013-07-29 13:41:37</FxDate>
             </Clearing>
         </Payment>
         <Processing code="VA.RF.90.00">
             <Timestamp>2013-07-29 13:41:37</Timestamp>
             <Result>ACK</Result>
             <Status code="90">NEW</Status>
             <Reason code="00">Successful Processing</Reason>
             <Return code="000.000.000">Transaction succeeded'</Return>
         </Processing>
    </Transaction>
</Response>
```

6.4.1 Partial Refunds (RF)

When only a part of the initially shipped products is returned, these products need to be specified in the request with the same CRITERION as were used in the Capture request. Multiple captures are supported; the Preauthorization remains valid until the full amount is captured.

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
<Transaction mode="LIVE" channel="xx" response="SYNC" source="XML">
       <Identification>
           <TransactionID>MerchantAssignedID</TransactionID>
           <ReferenceID>UniqueId of a Capture</ReferenceID>
       </Identification>
       <User login="xx" pwd="xx" />
       <Payment code="VA.RF">
              <Presentation>
                    <Amount>4.0</Amount>
                    <Currency>EUR</Currency>
              </Presentation>
```

```
        </Payment>
        <Analysis>
              <Criterion name="KLARNA_CART_ITEMSTOTAL">1</Criterion>
              <Criterion name="KLARNA_CART_ITEM1_ARTICLE_NR">1</Criterion>
              <Criterion name="KLARNA_CART_ITEM1_QUANTITY">4</Criterion>
        </Analysis>
</Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
<Transaction mode="LIVE" channel="xx" response="SYNC">
        <Identification>
              <TransactionID>MerchantAssignedID</TransactionID>
              <UniqueID>UniqueID</UniqueID>
              <ReferenceID>UniqueId of a Capture</ReferenceID>
              <ShortID>ShortID</ShortID>
        </Identification>
        <Payment code="VA.RF">
              <Clearing>
                    <Amount>4.00</Amount>
                    <Currency>EUR</Currency>
                    <Descriptor> ShortID Channelname Order Number 1234</Descriptor>
                    <FxRate>1.0</FxRate>
                    <FxSource>INTERN</FxSource>
                    <FxDate>2013-07-29 13:41:39</FxDate>
              </Clearing>
        </Payment>
        <Processing code="VA.RF.90.00">
              <Timestamp>2013-07-29 13:41:39</Timestamp>
              <Result>ACK</Result>
              <Status code="90">NEW</Status>
              <Reason code="00">Successful Processing</Reason>
              <Return code="000.000.000">Transaction succeeded'</Return>
         </Processing>
</Transaction>
</Response>
```

7

8 Registration Types

Registration types can be used for all payment methods. They are generally used for recurring transactions where data like account or customer information is registered for later retrieval in an initial Registration (RG) transaction. A registration transaction looks exactly like the corresponding Preauthorisation (PA), Debit (DB) or Credit (CD) transaction, except that the type is replaced by 'RG' and there is no Presentation tag.

After a successful Registration is complete, subsequent Preauthorisation (PA), Debit (DB), Credit (CD) or Receipt (RC) Transactions can reference the registered information. By submitting the 'registration' attributes within the Account or Customer tag, all subordinate tags that would be required for a normal PA, CD, or DB transaction can be omitted. The Unique ID of the registration response is used as registration reference in subsequent PA, DB, CD or RC transactions. A reference to an account registration would for example look like this:

```
<Account registration="678a456b789c123d456e789f012g432"/>
```

Note: Please note that a registration can be used as reference for up to 500 subsequent payment transactions (PA, DB, CD, RC, RV, RF).

Registration information can also be altered or disabled by submitting a Reregistration (RR) or Deregistration (DR) transaction. Registrations are either confirmed automatically (in most scenarios) or explicitly by a Confirmation (CF) transaction.

Instead of triggering a recurring PA, CD or DB individually, transactions that are predictable in amount and date – as in the case for subscriptions - can also be triggered by Scheduling (SD) transactions. Please refer to the chapter 'Job Group and Scheduling Types' for more information about automated Scheduling.

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

8.1 Registration (RG)

All following examples use the payment method direct debit (DD) for demonstration purposes. The payment methods credit card (CC) or credit transfer (CT) can also be used for registration.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="DD.RG"/>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
         <Country>DE</Country>
         <Limit>1000.00</Limit>
      </Account>
      <Customer>
         <Name>
            <Given>Joe</Given>
            <Family>Doe</Family>
         </Name>
         <Address>
            <Street>Leopoldstr. 1</Street>
            <Zip>80798</Zip>
            <City>München</City>
            <State>BY</State>
            <Country>DE</Country>
         </Address>
         <Contact>
            <Email>info@provider.com</Email>
            <Ip>123.123.123.123</Ip>
         </Contact>
      </Customer>
      <Frontend>
         < ResponseUrl >https://myshop.com/payment/registrationStatus.php</ResponseUrl>
      </Frontend>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="DD.RG.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <ConfirmationStatus>CONFIRMED</ ConfirmationStatus>
         <Reason code="00">Successful Processing</Reason>
```

```
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.RG"/>
      <Account>
         <Identification>Customer account specific identification number</Identification>
   <RegistrationUrl>https://test.ctpe.io/payment/registration?id=4028e4ee00a7bc470100a7bc
6a05000d</RegistrationUrl>
      </Account>
      <Connector>
         <Account>
            <Holder>Internal Account Holder</Holder>
            <Number>DE82700202700666898869</Number>
            <Bank>HYVEDEMM</Bank>
            <Country>DE</Country>
         </Account>
      </Connector>
   </Transaction>
</Response>
```

NB: The RG data are processed as follows:
The `<Account>`, `<Customer>` and `<Contact>` tag groups are stored, updated and referenced in the payment request as a whole. This means in case of a re-registration the whole tag group is replaced with the entirety of the data from the new tag group. For example, if a registration has five parameters in the `<Customer>` group and a re-registration takes place with only three parameters, the new registration has only three parameters in the `<Customer>` group. The other two parameters are erased.

8.2 Debit on Registration (DB)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="DD.DB">
         <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Account registration=" h987i654j321k0981765m432n210o987"/> <!—UniqueId of the
previous RG response -->
   </Transaction>
</Request>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="DD.DB.90.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="90">NEW</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code="000.000.000">Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DB">
         <Clearing>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
            <Descriptor>shop.de 1234.1234.1234 +49 (89) 12345 678 Order Number
1234</Descriptor>
            <Date>2003-02-13</Date>
            <Support>+49 (89) 1234 567</Support>
         </Clearing>
      </Payment>
   </Transaction>
</Response>
```

8.3 Reregistration (RR)

The Reference ID references the Unique ID of the original Registration.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="DD.RR"/>
      <Account>
         <Holder>Joe Doe</Holder>
         <Number>DE23100000001234567890</Number>
         <Bank>MARKDEF1100</Bank>
```

```
         <Country>DE</Country>
         <Limit>1000.00</Limit>
      </Account>
      <Frontend>
         <ResponseUrl>https://myserver.com/payment/confirmationCheck</ResponseUrl>
      </Frontend>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="DD.RR.00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <ConfirmationStatus>PENDING</ConfirmationStatus>
         <Redirect
url="https://test.ctpe.io/payment/ActiveRegistrationConfirmation.jsp?ctpe.tx.ID=
h987i654j321k098l765m432n210o987"/>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.RR"/>
   </Transaction>
</Response>
```

8.4 Deregistration (DR)

The Reference ID references to the Unique ID of the Registration transaction that is to be deregistered.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Payment code="DD.DR"/>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="DD.DR.00.05">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DR"/>
   </Transaction>
</Response>
```

9 Confirmation on Registration

If there is no Risk Management Process involved, Registrations are auto-confirmed with a Confirmation (CF) transaction and can therefore immediately be used to execute further payments.

In case of direct debit, there are a few countries that have documentation-based processes like France, Switzerland, Norway, Belgium and a few others. In this case a custom identification number is generated and needs to be confirmed explicitly in a second step. Therefore, the response also contains this ID and a URL for downloading the Registration confirmation document. In case of an account registration for paper based direct debit schemes, this URL would contain a paper document that must be signed by the shopper:

```
<Account>
   <Identification>Customer Account specific Identification Number</Identification>

   <RegistrationUrl>https://test.ctpe.io/payment/registration?id=4028e4ee00a7bc470100a7bc
   6a05000d</RegistrationUrl>
</Account>
```

As soon as a valid Registration confirmation document is received, a Confirmation (CF) transaction must be sent in. In the example of an account registration this document would be a signed paper document. For some countries like Norway the system generates a Confirmation request as soon as the clearinghouse has sent a notification.

Other processes return the paramater<RegistrationUrl>. This parameter contains a URL that the end user needs to visit (or be redirected to) to enter a confirmation code delievered by email, SMS or another medium.

If no explicit Confirmation is required for a specific Registration type, the system automatically generates a Confirmation.

Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

9.1 Confirmation (CF)

Confirmations are generated by the system depending on the configured registration mode of the channel.

By default a channel will be configured for auto-confirmation. This means that registration responses will automatically be CONFIRMED in the processing result group.

In some countries the Confirmation request is generated by the system as soon as the clearinghouse has received the Registration.

If the confirmation is handled asynchronously, notification will be sent via post response to the URL specified initially in the Registration request tag <ResponseUrl>.

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedIDofReferencedTransaction</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="DD.CF.00.05">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.CF"/>
   </Transaction>
</Response>
```

10 Scheduling Types (Automatic Recurrence)

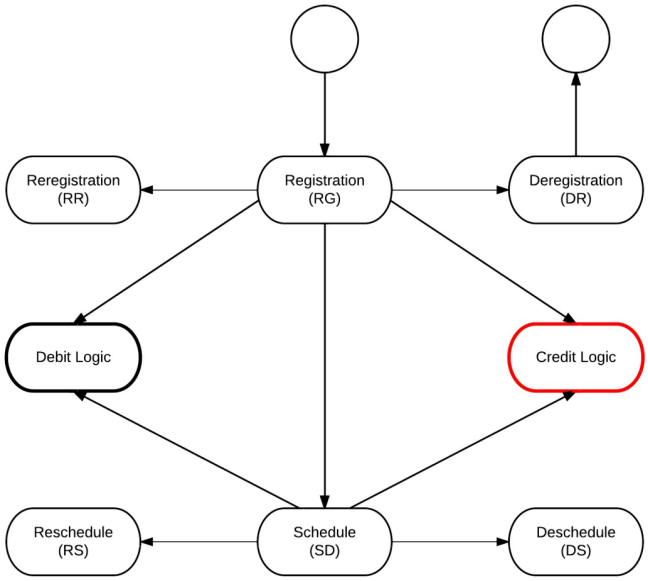
Payments which follow a certain predefined scheme, like subscriptions, can be scheduled in order to let the system automatically generate all upcoming transactions. Payments will continue until a Deschedule (DS) transaction is executed in which finalizes the Schedule. There are following scheduling types:

- Schedule (SD)
- Reschedule (RS)
- Deschedule (DS)

A Schedule (SD) transaction simply replaces the transaction type (like DB, CD or PA) in the payment code attribute with SD. This starts a Job that is described in the Job tag. The Job tag contains all parameters that are essential to define the Action, Execution Scheme, Cancellation Note and Duration of the Job. All scheduled transactions have the same Payment and Presentation tag as the initial Schedule transaction, only the Action/@type attribute - commonly DB, CD, or PA - replaces SD in the payment code attribute.

As all recurring transactions, a Schedule transaction needs to refer to a prior registration (RG) of the customer and account data to automatically generate transactions.

Note: Please note that a registration can be used as reference for up to 500 subsequent payment transactions (PA, DB, CD, RC, RV, RF).
Please refer to the Chapter 'Registration Types' for more information about the registration process.



Introductory remark:
When using the examples below, please replace the values of the following tags with your test or live parameters:

- SECURITY SENDER
- USER LOGIN
- USER PWD
- TRANSACTION CHANNEL

See the "Technical_Quick_Start" document for more details.

10.1 Schedule (SD)

All following examples use the payment method direct debit (DD) for demonstration purposes. Credit Card (CC) and Credit Transfer (CT) can also be used for scheduled payments. Please refer to the Chapter 'Job Group' and the document 'Recurrence Scenarios' for more examples and details.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
      </Identification>
      <Payment code="DD.SD">
         <Presentation>
            <Amount>10.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Job name="Trial Subscripton">
         <Action type="DB"/>
         <Execution>
            <Second>0</Second>
            <Minute>15</Minute>
            <Hour>10</Hour>
            <DayOfMonth>?</DayOfMonth>
            <Month>*</Month>
            <DayOfWeek>6L</DayOfWeek>
            <Year/>
         </Execution>
         <Notice>
            <Callable>DURATION_END</Callable>
            <Number>3</Number>
            <Unit>DAY</Unit>
         </Notice>
         <Duration>
            <Number>3</Number>
            <Unit>MONTH</Unit>
         </Duration>
```

```
      </Job>
      <Account registration="678a456b789c123d456e789f012g432"/>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.SD"/>
   </Transaction>
</Response>
```

10.2 Reschedule (RS)

A Reschedule request can be used to alter the payment and job parameters, while the job is running.

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>h987i654j321k098l765m432n210o987</ReferenceID>
      </Identification>
      <Payment code="DD.RS">
         <Presentation>
            <Amount>10.00</Amount>
            <Currency>EUR</Currency>
            <Usage>Order Number 1234</Usage>
         </Presentation>
      </Payment>
      <Job name="Trial Subscripton">
         <Action type="DB"/>
         <Execution>
            <Second>0</Second>
```

```
            <Minute>15</Minute>
            <Hour>10</Hour>
            <DayOfMonth>?</DayOfMonth>
            <Month>*</Month>
            <DayOfWeek>6L</DayOfWeek>
            <Year/>
         </Execution>
         <Notice>
            <Callable>ANYTIME</Callable>
            <Number>3</Number>
            <Unit>DAY</Unit>
         </Notice>
         <Duration>
            <Number>3</Number>
            <Unit>MONTH</Unit>
         </Duration>
      </Job>
      <Account registration="678a456b789c123d456e789f012g432"/>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>h987i654j321k098l765m432n210o987</ReferenceID>
      </Identification>
      <Processing code="00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.RS"/>
   </Transaction>
</Response>
```

10.3 Deschedule (DS)

Request:

```
<Request version="1.0">
   <Header>
      <Security sender="123a456b789c123d456e789f012g345" />
   </Header>
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
```

```
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <ReferenceID>h987i654j321k098l765m432n210o987</ReferenceID>
      </Identification>
      <Payment code="DD.DS"/>
   </Transaction>
</Request>
```

Response:

```
<Response version="1.0">
   <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
         <TransactionID>MerchantAssignedID</TransactionID>
         <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
         <ShortID>1234.5678.9876</ShortID>
         <ReferenceID>h987i654j321k098l765m432n210o987</ReferenceID>
      </Identification>
      <Processing code="DD.DS.00.00">
         <Timestamp>2003-02-12 14:58:07</Timestamp>
         <Result>ACK</Result>
         <Status code="00">SUCCESS</Status>
         <Reason code="00">Successful Processing</Reason>
         <Return code=„000.000.000“>Transaction succeeded</Return>
         <SecurityHash>2d0ec783cb7c2d5b117499e6211caef4</SecurityHash>
      </Processing>
      <Payment code="DD.DS"/>
   </Transaction>
</Response>
```

11 Transmission Options

11.1 HTTPS POST Interface

The HTTPS POST Interface is the simplest and fastest integration option available.

Authorization and Login

Each physical server requesting access to the HTTPS interface needs its own sender ID. The HTTPS gateway will serve requests only if it can validate the sender ID provided in the header of the xml message:

```
<Security sender="123a456b789c123d456e789f012g345" />
```

Please check with your account manager if you haven't received a sender ID.

It is especially important to track the user of a transaction for eTerminal transactions. For all automated requests a default system user is to be included:

```
<User login="421a456b789c123d456e789f012g098" pwd="56b789c123d456e789f"/>
```

Since HTTPS is a stateless protocol, the client application needs to resend the sender, login and password on each request by including the Security und User tag.

Please note that the XML message must be sent in a POST request as parameter 'load' and – of course – must be URL-encoded. The payment system requires the UTF-8 character set. The content-type of your message must be set to:

```
application/x-www-form-urlencoded;charset=UTF-8
```

In contrast to the sender ID, the channel ID is a purely logical subdivision of the business context and should not be confused with the sender ID. In other words, each sender ID can route several channel IDs and each channel ID can go over several sender IDs.

A merchant must successfully process transactions on the test gateway before going live. The link for downloading a test account is located in the 'Implementation Packages' document. Your account manager can also provide test accounts. Once you have completed your testing efforts please check with your account manager for approval to send transactions to the live gateway.

Test Gateway
URL : https://test.ctpe.io/payment/ctpe
Content-Type : `application/x-www-form-urlencoded;charset=UTF-8`

Live Gateway
URL : https://ctpe.io/payment/ctpe
Content-Type : `application/x-www-form-urlencoded;charset=UTF-8`

+++ end of document +++