



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

# Primerio Platform Documentation

## **XML Integrator (Queries)**

Release 3.2.1



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## Release History

Release	Description	Date	Changes
1.0.0	Initial Version	01-Nov-2004	Query interface for Chargebacks (CB), Confirmations (CF), Deregistrations (DR) and Receipts (RC)
[...]	[...]	[...]	[...]
3.0.0	Major Version	15-Dez-2010	Text revision and clarification
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	28-Oct-2011	Minor corrections
3.1.3	Minor Version	26-Mar-2012	Minor release – new BIN tag included
3.1.4	Minor Version	08-Mar-2013	Minor release
3.1.5	Minor Version	11-Mar-2014	Improved URLs
3.1.6	Minor Version	16-May-2014	Minor corrections
3.2.0	Major Version	11-Aug-2014	Reviewed, Improved URLs
3.2.1	Minor Version	20-11-2014	Minor corrections



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## **Preface**

### **PSP Platform Documentation - XML Integrator (Queries)**

Copyright © 2014 PrimerioPay - All rights reserved.  
Printed in Germany / European Union

The information contained in this document is intended only for the person or entity to which it is addressed and contains confidential and/or privileged material. Any review, retransmission, dissemination or other use of, or taking of any action in reliance upon, this information by persons or entities other than the intended recipient is prohibited. If you received this in error, please contact PrimerioPay and delete the material from any computer.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## Content

Release History .....	2
1 Introduction .....	5
2 Request Message .....	8
2.1 Request Encoding .....	8
2.2 Header Group .....	9
2.3 Query Group .....	9
2.4 User Group .....	12
2.5 Identification Group .....	12
2.5.1 Search for UniqueID or ShortID .....	13
2.5.2 Search for more than one UniqueID at the same time .....	13
2.5.3 Search for TransactionID .....	14
2.6 TransactionType Group .....	14
2.7 Period Group .....	15
2.8 Methods Group .....	15
2.9 Types Group .....	15
2.10 ProcessingResult Group .....	16
2.11 Account Group .....	16
3 Response Message .....	17
3.1 Result Group .....	17
3.2 Error Group .....	17
3.3 Transaction Group .....	17
3.4 Query response for transactions in process of completion .....	18
4 Typical Example Requests and Responses .....	19
4.1 Chargebacks (CB) .....	19
4.2 Confirmations (CF) .....	21
4.3 Deregistration (DR) .....	23
4.4 Receipts (RC) .....	24
4.5 Query Specific Transactions by ID .....	25
4.6 Query User Account (Login for User Accounts) .....	28
5 Error handling .....	30
6 Request - DTD .....	31
7 Response-DTD .....	32



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 1 Introduction

This document describes the interface for retrieving transaction status data from the Primerio system. It has been designed in a way that the retrieval and processing of the data can be fully scripted, or be used for manual pulls of data.

### URL

In order to retrieve the requested data from the Primerio system an XML message has to be sent via HTTP-POST within a single parameter called "load" to the following URL:

Test-URL: <https://test.ctpe.io/payment/query>

Live-URL: <https://ctpe.io/payment/query>

Upon receipt of a successful request, the system will return an XML stream of transactions and their respective data per the request criteria.

### Request Description

The XML Request version must be specified as "1.0".

The "Header" element is described in more detail in the XML Integrator (Transactions) documentation.

The "Query" describes the mode for which transactional data are to be received and for which entity of the hierarchy tree. Any defined entity (Channel, Merchant etc.) can be queried, thus, if a Channel is to be queried, specify that a Channel is being queried along with an accompanying ID.

The "Element" Type describes which transaction types you can query from the gateway. It is possible to query all available payment types. However, for some payment types a time limitation applies, ie they can only be queried over a period of one day.

At the moment five transaction types are available for searches over a maximum period of one month:

- Chargeback (CB)
- Chargeback Reversal (CR)
- Confirmation (CF)
- Deregistration (DR)
- Receipt (RC)

In those cases the difference between "from" and "to" in the Period tag must be lower than or equal to one month

All other types can only be queried over a period of one day, meaning "from" and "to" in the Period tag must



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

be the same.

**IMPORTANT:** The maximum number of transactions per query is limited depending on the types and period you are querying for:

- The maximum number of results per query in general is 10000.
- Should a time period longer than a week or the time period not be specified, the maximum number of results will be 2000.
- Should a type other than CB, CR, CF, DR, or RC be queried, the maximum number is 2000.
- Should "TransactionID" be searched, the maximum number is 200.
- Queries for ShortID and UniqueID do not require any time period or other additional query parameters
- If the time period is lower than ten minutes, the query can be executed without any limiting query parameters

For system stability and security reasons, Primerio reserves the right to block the query interface for single merchants should the Query API receive unusual load or usage patterns. See chapter 4 for typical use cases and recommended queries.

### Simple Request Example

```
<Request version="1.0">
  <Header>
    <Security sender="ff80808109c5bcc00109c5bce9f1003a"/>
  </Header>
  <Query entity="ff80808109c5bcc00109c5bce9f50056" level="CHANNEL"
mode="INTEGRATOR_TEST" type="STANDARD">
    <User login="ff80808109c5bcc00109c5bce9f20042" pwd="geheim"/>
    <Period from="2006-03-04" to="2006-03-04"/>
    <Types>
      <Type code="RF"/>
      <Type code="PA"/>
      <Type code="RV"/>
    </Types>
  </Query>
</Request>
```

### Simple Response Example 1

```
<Response version="1.0">
  <Result response='SYNC' type="LIST">
    <Transaction .../>
    <Transaction .../>
    <Transaction .../>
    ...
  </Result>
</Response>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## Simple Response Example 2

```
<Response>  
  <Error>  
    <Timestamp>2006-03-04 15:50:35</Timestamp>  
    <Return code="100.200.101">invalid period, if searching for non-backchannel-types  
(CB, RC, RL, CF, DR) from and to must be on the same day</Return>  
  </Error>  
</Response>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 2 Request Message

### 2.1 Request Encoding

For all payment requests containing shopping and possibly payment data, the request header must contain the Content-Type / charset parameter with the charset encoding set to "UTF-8". The actual content's 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**

Example headers:

**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:**

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





PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 2.2 Header Group

The header group of the XML request 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 an 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 as received from an account manager or support.

## 2.3 Query Group

```
<Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"  
type="STANDARD" maxCount="5">
```

The Query tag has three attributes which determine the processing of the query.

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.



PrimeiroPay S.a.r.l.  
 7 r. Robert Stuemper  
 2557 Luxembourg  
 Luxembourg

Value for level	Description
CHANNEL	Transactions are queried on channel level (ID specified with entity is a channel ID)
MERCHANT	Transactions are queried on merchant level (ID specified with entity is a merchant ID)
PSP	Transactions are queried on PSP level (ID specified with entity is a PSP ID)
Value for type	Description
STANDARD	This is the default type. No special behavior activated.
ACTIVE_TRANSACTIONS	Queries all transactions which match the given query parameters and which are active transactions of their sessions.
LINKED_TRANSACTIONS	Queries all transactions which are linked directly or indirectly to one specific transaction. For this type the ID or ShortID of one transaction must be specified.
AVAILABLE_TRANSACTIONS	Queries all transactions which match the given query parameters and which are available transactions of their sessions. The difference between ACTIVE_TRANSACTIONS and AVAILABLE_TRANSACTIONS applies to registrations only: "active" are only confirmed registrations, "available" are not confirmed registrations also.
ACTIVE_LINKED_TRANSACTIONS	Queries all transactions which are linked directly or indirectly to one specific transaction and which are active transactions in their session. For this type the ID or ShortID of one transaction must be specified.
AVAILABLE_LINKED_TRANSACTIONS	Queries all transactions which are linked directly or indirectly to one specific transaction and which are active transactions in their session. For this type the ID or ShortID of one transaction must be specified. The difference between ACTIVE_LINKED_TRANSACTIONS and AVAILABLE_LINKED_TRANSACTIONS applies to registrations only: "active" are only confirmed registrations, "available" are not confirmed registrations also.
Value for maxCount	Description
Number	Maximum number of returned transactions by the query. This is a useful parameter to get, for example, the last X transactions in a given query use case.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

Value for entity	Description
Alphanumeric 32	ID of the entity specified in level. The entity ID is a unique key for the identification of the unit that sends transactions into the system.

In case you want to query SCHEDULER generated transactions only (See Documents "XML\_Transactions", Chapter Recurrence and "Recurrence Scenarios" for details) use the additional attribute "source":

```
<Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432" type="STANDARD" source="SCHEDULER">
```

Value for source	Description
SCHEDULER	Retrieve only transactions that were system generated by the internal scheduler (recurrent payment)



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 2.4 User Group

The User tag group contains security related information. Check the document "Technical Quickstart" for information on how to retrieve this login data for an account.

```
<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 3..32	A password which fits the login UID has to be provided. It is distributed together with the login UID.

## 2.5 Identification Group

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

- Unique ID
- Short ID
- TransactionID
- BulkID

```
<Identification>  
<UniqueID>12345678901234567890123456789012</UniqueID>  
<ShortID>1234.1234.1234</ShortID>  
<TransactionID>Merchant_Defined_ID_For_A_Transaction</TransactionID>  
<BulkID>Batch12</BulkID>  
</Identification>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

### 2.5.1 Search for UniqueID or ShortID

This group allows the user to query for specific transactions if the ShortID or the UniqueID of the transaction is known. Both IDs are part of each XML Transaction Response. Only one of the two IDs is required. If either UniqueID or ShortID is present, Period and Types are optional.

Identification Tag	Data Type	Length	Mandatory / Optional	Description
UniqueID	Alphanumeric	32	Conditionally Mandatory	System-generated 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	Conditionally Mandatory	ID which is used for manual entry and search purposes. The likelihood for uniqueness is very high, but not guaranteed.

### 2.5.2 Search for more than one UniqueID at the same time

This group allows the user to query for specific transactions the UniqueIDs are known and return them all at once. Period and Types are optional.

Identification Tag	Data Type	Length	Mandatory / Optional	Description
UniqueIDs	Alphanumeric	32	Conditionally Mandatory	Contains a list of <ID> sub elements
ID	Alphanumeric	32	Conditionally Mandatory	System-generated ID where the uniqueness within the system is absolutely guaranteed. Has to be used for all automated matching and reference purposes.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

### Example:

```
<Identification>  
  <UniqueIDs>  
    <ID>1234567890123456789012bb3456789012</ID>  
    <ID>ff8081811bfa5356011bfbe22a2800b4</ID>  
  </UniqueIDs>  
</Identification>
```

### 2.5.3 Search for TransactionID

This group also allows the user to query for specific transactions by the TransactionID. This is the merchant-assigned ID of a transaction.

Identification Tag	Data Type	Length	Mandatory / Optional	Description
TransactionID	Alphanumeric	32	Optional	ID the merchant has assigned to a specific payment transaction and submitted with the original payment transaction.

**IMPORTANT:** This ID does not necessarily have to be unique when it gets submitted the payment system with a payment transaction. Nevertheless it is highly recommended to use unique (or at least selective) ones. For safety and stability reasons of the query interface, this query allows you to retrieve a maximum of 200 transactions per TransactionID.

### 2.6 TransactionType Group

The TransactionType group contains an alternative and/or extension to the Methods group. It allows you to define what types of transactions your query result should contain.

```
<TransactionType>PAYMENT</TransactionType>
```

Value of TransactionType	Description
PAYMENT	Payment Transaction Types. Those are CB, CD, CR, DB, CP, PA, RB, RC, RF and RV
REGISTER	Registration Transaction Types. Those are CF, DR, RG and RR
SCHEDULE	Scheduling Transaction Types. Those are DS, RS and SD
RISKMANAGEMENT	Risk Management Transaction Types. Those are EA, RI, 3D, SA, and IC.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 2.7 Period Group

This tag group specifies the date period of the query. The "from" date is always extended with the time 00:00:00 and the "to" date with 23:59:59. All times are 24-hour format UTC time zones only.

For the following payment types the "from" and "to" date can be any day:

- Chargeback (CB)
- Chargeback Reversal (CR)
- Confirmation (CF)
- Deregistration (DR)
- Receipt (RC)

For all other payment types the "from" and "to" date have to be on the same day. This means only one day can be queried within one XML query request.

```
<Period from="2004-09-01" to="2004-09-15 12:00:00"/>
```

Value for from	Description
Date, format yyyy-MM-dd [HH:mm:ss]	Date from when the query starts. UTC 24-hour time zone.
Value for to	Description
Date, format yyyy-MM-dd [HH:mm:ss]	Date when the query ends. UTC 24-hour time zone.

## 2.8 Methods Group

The Methods tag allows for filtering by payment method.

```
<Methods>  
  <Method code="CC"/>  
  <Method code="DD"/>  
</Methods>
```

Value for code	Description
Alpha(2)	A valid payment method. For a complete list of payment methods refer to the document "XML Transactions"

## 2.9 Types Group

The Types tag group allows for filtering by a valid payment type.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Types>  
  <Type code="CB"/>  
  <Type code="CF"/>  
</Types>
```

Value for code	Description
Alpha(2)	A valid payment type. For a complete list of payment types refer to the document "XML Transactions"

## 2.10 ProcessingResult Group

The ProcessingResult tag group provides functionality for filtering by success or failure of a transaction. It is the same tag received in an XML payment response as part of the Processing Tag Group (See document "XML Transactions").

```
<ProcessingResult>ACK</ProcessingResult>
```

Value for ProcessingResult	Description
ACK	Get successful Transactions only
NOK	Get failed (rejected) Transactions only

## 2.11 Account Group

The Account group enables searching by user account. This tag group is only needed for searching for registered user accounts.

```
<Account>  
  <Id>joe.doe@email.net</Id >  
  <Password>pwd12345</Password>  
  <Brand>TESTWALLET</Brand>  
</Account>
```

Tag Identification	of	Data Type	Length	Mandatory / Optional	Description
Id		Alphanumeric	128	Mandatory	ID of the user with the User Account
Brand		Alphanumeric	32	Mandatory	Brand of the User Account
Password		Password	32	Mandatory	Password of the registered user





PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

### 3 Response Message

The result to a query request message is – like the response to an XML transaction request – surrounded by a <Response> tag. The response tag contains either a Result Group or, in case of an invalid request, an Error Group.

#### 3.1 Result Group

```
<Result response='SYNC' type="LIST">
```

Value for response	Description
SYNC	All requested transactions are part of this request and are nested inside the Result Group
Value for type	Description
LIST	Specifies that the transactions are provided as a list inside the Result group

#### 3.2 Error Group

```
<Error>  
  <Timestamp>2004-11-11 11:11:11</Timestamp>  
  <Return code="100.100.101">invalid entity type</Return>  
</Error>
```

Tag of Error	Description
Timestamp	Date and Time of the request, format is "yyyy-MM-dd hh:mm:ss"
Return	Contains error information describing the occurred error as text
Value for code	Description
Alphanumeric, format is nnn.nnn.nnn	Return code identifying the error that occurred. For a list of all error codes check <a href="https://test.ctpe.net/payment/codes/queryErrorCodes.jsp">https://test.ctpe.net/payment/codes/queryErrorCodes.jsp</a>

#### 3.3 Transaction Group

Check the document "XML Transactions" for more details about the structure of the Transaction group.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

### 3.4 Query response for transactions in process of completion

Transactions for which the processing isn't completed yet at the time of the query execution the query response will return the following status:

```
<Processing code="CC.PA.90.00">  
  <ConnectorDetails/>  
  <Reason code="00">Successful Processing</Reason>  
  <Result>ACK</Result>  
  <Return code="000.000.000">Transaction succeeded</Return>  
  <Status code="90">NEW</Status>  
</Processing>
```

This of course could mislead the merchant to the assumption that the transaction is successful and authorized, which it's not at this point.

To avoid confusion and eventual financial loss please consider the following steps before accepting transactions with 000.000.000 / success return code:

- Only submit a query, if after 30 seconds the HTTP response of a transaction hasn't been received.
- Check in the query response for the tag <ConnectorDetails>. If there is none, the 000.000.000 / success return code should not be accepted but a new query should be sent after a 5-10 seconds.
- Additionally, check in the query response for the elements <Identification> and <Clearing> and their sub-elements. If there is none, the 000.000.000 / success return code should not be accepted but a new query should be sent after a 5-10 seconds.
- In case risk management is used, check in the query response for RM.RI transaction data. If there is none, the 000.000.000 / success return code should not be accepted but a new query should be sent after a 5-10 seconds.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 4 Typical Example Requests and Responses

### 4.1 Chargebacks (CB)

For retrieving chargebacks for a certain period a request has to be sent to the query engine at:

Test-URL: <https://test.ctpe.io/payment/query>

Live-URL: <https://ctpe.io/payment/query>

#### **Typical Use Case: Automatically retrieve chargebacks from the payment system.**

Depending on the acquirer or bank, new Chargebacks usually appear once a day in the system. For details contact your account manager. It is therefore recommended to limit chargeback requests to a few times a day, depending on the schedule which a payment connector provides its chargeback data.

Please note: For system safety and stability reasons the merchant query access may be blocked in case of receiving too many automated requests from this account within a short timeframe.

```
<Request version="1.0">
  <Header>
    <Security sender="123a456b789c123d456e789f012g345"/>
  </Header>
  <Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"
type="STANDARD">
    <User login="peter" pwd="thewolf"/>
    <Period from="2004-09-01" to="2004-09-07"/>
    <Types>
      <Type code="CB"/>
    </Types>
  </Query>
</Request>
```

This request will retrieve the chargebacks for the channel identified by the ID 678a456b789c123d456e789f012g432 for the week of the 1st of September to the 7th of September. Assuming two chargebacks have been issued during the aforementioned period, the following will be returned:

```
<Response version="1.0">
  <Result response="SYNC" type="LIST">
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
        <TransactionID>4711</TransactionID>
        <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
      </Identification>
    </Transaction>
  </Result>
</Response>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<ShortID>1234.5678.9876</ShortID>
<ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
</Identification>
<Processing code="DD.CB.90.00">
  <Timestamp>2014-11-13 08:19:52</Timestamp>
  <Result>ACK</Result>
  <Status code="90">NEW</Status>
  <Reason code="00">Successful Processing</Reason>
  <Return code="000.100.206">Revocation or Dispute</Return>
</Processing>
<Account>
  <Number>*****6581</Number>
  <Bin>456789</Bin>
  <Holder>bob kosel</Holder>
  <Bank>38050000</Bank>
  <Country>DE</Country>
</Account>
<Payment code="DD.CB">
  <Presentation>
    <Amount>1.00</Amount>
    <Currency>EUR</Currency>
  </Presentation>
  <Clearing>
    <Amount>1.00</Amount>
    <Currency>EUR</Currency>
    <Date>2004-09-02</Date>
  </Clearing>
</Payment>
</Transaction>
<Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
  <Identification>
    <TransactionID>4712</TransactionID>
    <UniqueID>h987i654j321k098l765m43333222111</UniqueID>
    <ShortID>1234.5678.8191</ShortID>
    <ReferenceID>m123n456o789p876q543r21111222333</ReferenceID>
  </Identification>
  <Processing code="DD.CB.00.00">
    <Timestamp>2004-09-07 14:58:07</Timestamp>
    <Result>ACK</Result>
    <Status code="90">NEW</Status>
    <Reason code="00">Successful Processing</Reason>
    <Return code="000.100.206">Revocation or Dispute</Return>
  </Processing>
  <Account>
    <Number>*****6581</Number>
    <Bin>456789</Bin>
    <Holder>bob kosel</Holder>
    <Bank>38050000</Bank>
    <Country>DE</Country>
  </Account>
  <Payment code="DD.CB">
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Presentation>
  <Amount>3.00</Amount>
  <Currency>EUR</Currency>
</Presentation>
<Clearing>
  <Amount>3.00</Amount>
  <Currency>EUR</Currency>
  <Date>2004-09-07</Date>
</Clearing>
</Payment>
</Transaction>
</Result>
</Response>
```

## 4.2 Confirmations (CF)

### Typical Use Case: Automatically retrieve confirmations from the payment system.

For processes where a Registration (RG) is not auto-confirmed (because of further risk management processes or because of a mandate based direct debit process) Confirmations can be issued anytime after the Registration is called.

Confirmations are for example the only way to see if a mandate-based direct debit process has been registered by the client. Transactions can only be issued after a confirmation is received. Thus, confirmations are typically queried once per day.

```
<Request version="1.0">
  <Header>
    <Security sender="123a456b789c123d456e789f012g345"/>
  </Header>
  <Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"
type="STANDARD">
    <User login="peter" pwd="thewolf"/>
    <Period from="2004-09-08" to="2004-09-08"/>
    <Types>
      <Type code="CF"/>
    </Types>
  </Query>
</Request>
```

This request will retrieve the confirmations for the channel identified by the ID 678a456b789c123d456e789f012g432 for the 8th of September. Assuming three confirmations have been issued during the aforementioned period, the following will be returned:



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Response version="1.0">
  <Result response="SYNC" type="LIST">
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
        <TransactionID>reg 0815</TransactionID>
        <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
        <ShortID>1234.5678.9876</ShortID>
      </Identification>
      <Processing code="DD.CF.00.00">
        <Timestamp>2004-09-08 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>
      </Processing>
      <Payment code="DD.CF"/>
      <Account id="678a456b789c123d456e789f0123333"/>
    </Transaction>
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
        <TransactionID>reg 0816</TransactionID>
        <UniqueID>h987i654j321k098l765m432n2888999</UniqueID>
        <ShortID>1234.5678.0203</ShortID>
      </Identification>
      <Processing code="DD.CF.00.00">
        <Timestamp>2004-09-08 14:58:23</Timestamp>
        <Result>ACK</Result>
        <Status code="00">SUCCESS</Status>
        <Reason code="00">Successful Processing</Reason>
        <Return code="000.000.000">Transaction succeeded</Return>
      </Processing>
      <Payment code="DD.CF"/>
      <Account id="678a456b789c123d456e789f0124545"/>
    </Transaction>
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
        <TransactionID>reg 0817</TransactionID>
        <UniqueID>h987i654j321k098l765m432n2321321</UniqueID>
        <ShortID>1234.5678.0198</ShortID>
      </Identification>
      <Processing code="DD.CF.00.00">
        <Timestamp>2004-09-08 14:58:55</Timestamp>
        <Result>ACK</Result>
        <Status code="00">SUCCESS</Status>
        <Reason code="00">Successful Processing</Reason>
        <Return code="000.000.000">Transaction succeeded</Return>
      </Processing>
      <Payment code="DD.CF"/>
      <Account id="678a456b789c123d456e789f07744411"/>
    </Transaction>
  </Result>
</Response>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
</Result>  
</Response>
```

Check for the following in your system in order to match the confirmation responses to appropriate transactions: Response / Transaction / Account / @id.

### 4.3 Deregistration (DR)

#### Typical Use Case: Automatically retrieve deregistered registrations from the payment system.

The deregistration process of an account can happen in two ways:

- The client uses your system to do so and your system triggers a deregistration request
- The possibility also exists that the end-customer notifies his or her bank to cease payments

The Primerio system polls regularly for newly cancelled mandates on the banks side. If a mandate has been cancelled externally, a deregistration transaction is created that can be retrieved via the query interface. All transactions that reference a deregistered account will be declined by the gateway. Daily polling for deregistration transactions is recommended.

```
<Request version="1.0">  
  <Header>  
    <Security sender="123a456b789c123d456e789f012g345"/>  
  </Header>  
  <Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"  
type="STANDARD">  
    <User login="peter" pwd="thewolf"/>  
    <Period from="2004-09-08" to="2004-09-08"/>  
    <Types>  
      <Type code="DR"/>  
    </Types>  
  </Query>  
</Request>
```

This request will retrieve the deregistrations for the channel identified by the ID 678a456b789c123d456e789f012g432 for the 8th of September. Assuming one confirmation has been issued during the aforementioned period, the following will be returned:

```
<Response version="1.0">  
  <Result response="SYNC" type="LIST">  
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">  
      <Identification>  
        <TransactionID>reg 0815</TransactionID>  
        <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>  
        <ShortID>1234.5678.9876</ShortID>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
</Identification>
<Processing code="DD.DR.00.00">
  <Timestamp>2004-09-08 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>
</Processing>
<Payment code="DD.DR"/>
<Account id="678a456b789c123d456e789f0123333"/>
</Transaction>
</Result>
</Response>
```

Check for the following in your system in order to match the deregistration responses to appropriate transactions: Response / Transaction / Account / @id.

#### 4.4 Receipts (RC)

**Typical Use Case: Automatically retrieve if somebody has transferred money into a bank account.**

Receipts need to be retrieved in order to confirm if a prepayment (PP) has been fulfilled.

```
<Request version="1.0">
  <Header>
    <Security sender="123a456b789c123d456e789f012g345"/>
  </Header>
  <Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"
type="STANDARD">
    <User login="peter" pwd="thewolf"/>
    <Period from="2004-09-01" to="2004-09-07"/>
    <Types>
      <Type code="RC"/>
    </Types>
  </Query>
</Request>
```

This request will retrieve the receipts for the channel identified by the ID 678a456b789c123d456e789f012g432 for the week of the 1st of September to the 7th of September.

Assuming one receipt has been issued during the aforementioned period, the following will be returned:

```
<Response version="1.0">
  <Result response="SYNC" type="LIST">
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
```





PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<TransactionID>4711</TransactionID>
<UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
<ShortID>1234.5678.9876</ShortID>
<ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
</Identification>
<Processing code="DD.RC.00.00">
  <Timestamp>2004-09-02 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>
</Processing>
<Account>
  <Number>*****6581</Number>
  <Bin>456789</Bin>
  <Holder>bob kosel</Holder>
  <Bank>38050000</Bank>
  <Country>DE</Country>
</Account>
<Payment code="DD.RC">
  <Presentation>
    <Amount>1.00</Amount>
    <Currency>EUR</Currency>
  </Presentation>
  <Clearing>
    <Amount>1.00</Amount>
    <Currency>EUR</Currency>
    <Date>2004-09-02</Date>
  </Clearing>
</Payment>
</Transaction>
</Result>
</Response>
```

## 4.5 Query Specific Transactions by ID

### Typical Use Case: Automatically retrieve transactions from the payment system where you only have an ID

In order to retrieve a transaction where the UniqueId or ShortId is known, only one transaction can be queried. For example this can be particularly useful to retrieve a Registration of a user to be able to show the user the (masked) payment data or customer data entered the last time the user paid in the shop. Another possibility is to use the merchant-submitted <TransactionID> for the query. This is very helpful in case the UniqueID or ShortID are no longer available or were never it (e.g. Timeout).

```
<Request version="1.0">
  <Header>
    <Security sender="123a456b789c123d456e789f012g345"/>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
</Header>
<Query mode="LIVE" level="CHANNEL" entity="678a456b789c123d456e789f012g432"
type="STANDARD">
  <User login="peter" pwd="thewolf"/>
  <Period from="2007-01-01" to="2007-01-07"/>
  <Identification>
    <ShortID>1234.1230.4422</ShortID>
  </Identification>
</Query>
</Request>
```

#### Typical response message for a Receipt:

```
<Response version="1.0">
  <Result response="SYNC">
    <Transaction mode="LIVE" response="SYNC" channel="678a456b789c123d456e789f012g432">
      <Identification>
        <TransactionID>4711</TransactionID>
        <UniqueID>h987i654j321k098l765m432n210o987</UniqueID>
        <ShortID>1234.5678.9876</ShortID>
        <ReferenceID>m123n456o789p876q543r210s123t456</ReferenceID>
      </Identification>
      <Processing code="DD.RC.00.00">
        <Timestamp>2004-09-02 14:58:07</Timestamp>
        <Result>ACK</Result>
        <Status code="00">SUCCESS</Status>
        <Reason code="40">Revocation or Dispute</Reason>
        <Return code="000.200.000">Transaction succeeded</Return>
      </Processing>
      <Payment code="DD.RC">
        <Presentation>
          <Amount>1.00</Amount>
          <Currency>EUR</Currency>
        </Presentation>
        <Clearing>
          <Amount>1.00</Amount>
          <Currency>EUR</Currency>
          <Date>2004-09-02</Date>
        </Clearing>
      </Payment>
    </Transaction>
  </Result>
</Response>
```

#### Typical response message for a Debit:

```
<Response>
  <Result response="SYNC" count="1">
    <Transaction mode="INTEGRATOR_TEST" channel="ff80808112fc2f530112fc2f63e7000f"
response="SYNC" source="XML">
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Identification>
  <ShortID>9663.7985.4310</ShortID>
  <UniqueID>ff80808112f795b00112fc2f7859003f</UniqueID>
  <TransactionID>1174</TransactionID>
</Identification>
<Relevances />
<Payment code="CC.PA">
  <Clearing>
    <Amount>73.50</Amount>
    <Currency>EUR</Currency>
    <Descriptor>9663.7985.4310 TPX_order#           222
PSP_A/MER_A/DEFAULT</Descriptor>
    <FxRate>1.0</FxRate>
    <FxSource>INTERN</FxSource>
    <FxDate>2006-08-05 16:00:08</FxDate>
  </Clearing>
  <Presentation>
    <Amount>73.50</Amount>
    <Currency>EUR</Currency>
    <Usage>TPX_order#           222</Usage>
  </Presentation>
</Payment>
<Account>
  <Number>*****1881</Number>
  <Bin>456789</Bin>
  <Holder>bob kosel</Holder>
  <Brand>VISA</Brand>
  <Expiry month="12" year="2014" />
</Account>
<Customer>
  <Name>
    <Family>kosel</Family>
    <Given>bobby</Given>
    <Company>kosel co.</Company>
    <Salutation>MR</Salutation>
    <Title></Title>
  </Name>
  <Contact>
    <Email>bob_kosel@googlemail.com</Email>
    <Ip>101.202.011.022</Ip>
    <Mobile>0049 199 6542123</Mobile>
    <Phone>0049 199 6542123</Phone>
  </Contact>
  <Address>
    <City>Frankfurt</City>
    <Country>DE</Country>
    <State>DE7</State>
    <Street>Hauptstrasse</Street>
    <Zip>61821</Zip>
  </Address>
</Customer>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Processing code="CC.PA.90.00">
  <Timestamp>2006-08-05 16:00:08</Timestamp>
  <Result>ACK</Result>
  <Status code="90">NEW</Status>
  <Reason code="00">Successful Processing</Reason>
  <Return code="000.100.110">Request successfully processed in 'Merchant in
Integrator Test Mode'</Return>
</Processing>
  <RequestTimestamp>2006-08-05 16:00:08</RequestTimestamp>
  <Analysis />
</Transaction>
</Result>
</Response>
```

#### 4.6 Query User Account (Login for User Accounts)

##### Typical Use Case: Automatically login a wallet user.

In order to retrieve the User Account registration of a user by ID (Username) and Password, you need to specify the Account tag group with ID, Password and Brand.

```
<?xml version='1.0' encoding='utf-8'?>
<Request version='1.0'>
  <Query entity='ff8080811077e476011077e6419c00fd' level='CHANNEL'
mode='INTEGRATOR_TEST' type="ACTIVE_TRANSACTIONS">
  <User login='ff8080811077e476011077e6419900fa' pwd='demo' />
  <Account>
    <Id>test</Id>
    <Password>test</Password>
    <Brand>TESTUACCBRAND</Brand>
  </Account>
</Query>
<Header>
  <Security sender='ff8080811077e476011077e6419800f8' type='MERCHANT' />
</Header>
</Request>
```

Typical response message:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Response>
  <Result response="SYNC" count="1">
    <Transaction mode="INTEGRATOR_TEST" channel="ff8080811077e476011077e6419c00fd"
response="SYNC" source="XML">
      <Identification>
        <ShortID>2397.9030.7378</ShortID>
        <UniqueID>ff808081110d897f01110d8e84a10002</UniqueID>
        <TransactionID>demosp</TransactionID>
      </Identification>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

```
<Payment code="UA.RG"/>
<Account>
  <Id>*****test</Id>
  <Holder>student tester</Holder>
  <Brand>TESTUACCBRAND</Brand>
</Account>
<Customer>
  <Name>
    <Family>tester</Family>
    <Given>student</Given>
    <Company/>
    <Salutation>MR</Salutation>
    <Title/>
  </Name>
  <Contact>
    <Email>user@somewhere.org</Email>
    <Ip>123.123.123.123</Ip>
    <Mobile>+49-181-7654321</Mobile>
    <Phone>+49-112-11119999</Phone>
  </Contact>
  <Address>
    <City>Stadt</City>
    <Country>ZA</Country>
    <State/>
    <Street>Street 3</Street>
    <Zip>80798</Zip>
  </Address>
  <Details>
    <Identity paper="IDCARD">xyz123</Identity>
  </Details>
</Customer>
<Processing code="UA.RG.90.00">
  <Timestamp>2007-03-01 13:51:55</Timestamp>
  <Result>ACK</Result>
  <Status code="90">NEW</Status>
  <Reason code="00">Successful Processing</Reason>
  <Return code="000.100.110">Request successfully processed in 'Merchant in
Integrator Test Mode'</Return>
</Processing>
<RequestTimestamp>2007-03-01 13:51:54</RequestTimestamp>
<Analysis>
  <Criterion name="jobtitle">manager</Criterion>
  <Criterion name="gender">male</Criterion>
  <Criterion name="age">20</Criterion>
</Analysis>
</Transaction>
</Result>
</Response>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 5 Error handling

If a query is rejected due to an error the Response will contain an Error-Tag. The following example shows a request that is declined because the type is unknown (please be aware that codes and messages are subject to change):

```
<Request version="1.0">
  <Header>
    <Security sender="123a456b789c123d456e789f012g345"/>
  </Header>
  <Query mode="LIVE" level="MERCHANT" entity="678a456b789c123d456e789f012g432
type="STANDARD">
    <User login="peter" pwd="thewolf"/>
    <Period from="2004-09-08 12:00:00" to="2004-09-08 13:00:00"/>
    <Types type="LIST">
      <Type code="MM"/>
    </Types>
  </Query>
</Request>
```

This request will yield an error message:

```
<Response version="1.0">
  <Error>
    <Timestamp>2004-11-11 11:11:11</Timestamp>
    <Return code="100.100.101">invalid entity type</Return>
  </Error>
</Response>
```

For a current list of all possible return codes see <https://test.ctpe.net/payment/codes/queryErrorCodes.jsp>.



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 6 Request - DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Request (Header, Query)>
<!ATTLIST Request
  version (1.0) #REQUIRED
>
<!ELEMENT Header (Security)>
<!ELEMENT Security EMPTY>
<!ATTLIST Security
  sender CDATA #REQUIRED
>
<!ELEMENT Query (User, Period?, Identification?, Types?)>
<!ATTLIST Query
  mode (LIVE | CONNECTOR_TEST | INTEGRATOR_TEST) #REQUIRED
  level (CHANNEL | MERCHANT | PSP) #REQUIRED
  entity CDATA #REQUIRED
>
<!ELEMENT User EMPTY>
<!ATTLIST User
  login CDATA #REQUIRED
  pwd CDATA #REQUIRED
>
<!ELEMENT Period EMPTY>
<!ATTLIST Period
  from CDATA #REQUIRED
  to CDATA #REQUIRED
>
<!ELEMENT Identification (UniqueID?, ShortID?)>
<!ELEMENT ShortID (#PCDATA)>
<!ELEMENT UniqueID (#PCDATA)>
<!ELEMENT Types (Type+)>
<!ELEMENT Type EMPTY>
<!ATTLIST Type
  code CDATA #REQUIRED
>
```



PrimeiroPay S.a.r.l.  
7 r. Robert Stuemper  
2557 Luxembourg  
Luxembourg

## 7 Response-DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Response (Result | Error)>
<!ATTLIST Response
  version CDATA #IMPLIED
>
<!ELEMENT Result (Transaction*)>
<!ATTLIST Result
  response (SYNC) #IMPLIED
  count CDATA #IMPLIED
  type (LIST) #IMPLIED
>
<!-- for full definition of the Transaction element refer to the XML Transaction
documentation-->
<!ELEMENT Transaction (#PCDATA)>
<!ATTLIST Transaction
  response (SYNC | ASYNC) #IMPLIED
  mode CDATA #IMPLIED
  channel CDATA #IMPLIED
>

<!ELEMENT Error (Timestamp, Return)>
<!ELEMENT Timestamp (#PCDATA)>

<!ELEMENT Return (#PCDATA)>
<!ATTLIST Return
  code CDATA #REQUIRED
>
```