Íslandsbanki

Vefþjónusta

Sambankaskema

10. mars 2008

TABLE OF CONTENTS

Design

Introduction

Purpose

The main purpose of this document is to give an overview of services which make it possible for users connect to a banking institution and perform financial operations.
The intended users of this document:
Designers and programmers at banking institutions which will design and program systems along with unit and integration tests on the services described here.
IT Workers at banking institutions, which will deploy and maintain the services.
Companies, ISV's or independent programmers, creating systems utilizing these services.

Scope

This document attempts to address issues of security, the standards that are adhered to and the main design decisions made about how the services behave. The operations are described in details.
This document does not go into service implementation details which will inevitably be different for different banks. Neither is it the purpose of this document to describe details about client implementation.

References

WS-Security Specification
Technical specification of HTTP over TLS (RFC 2818)

Design Goals and Limitations

Problem statement

Today all Icelandic banks provide roughly the same financial services over the Internet or closed nets. The services enable users to retrieve bank statements, transfer funds and manipulate claims. The methods and protocols of communication vary from sending text based documents over FTP, XML over HTTP to SOAP message interchanges. Because of this users must customize their system to each banking institution.
By standardizing the interface to these financial services, users can create one client which can communicate with all institutions which implement the services.

Design Goals

The main design goals that were set at the start of the project were:
Communication with the services should be easy and inexpensive.

1. It should be easy for users/programmers to create clients which can be used to interact with the services.
2. Build on known and accepted standards as much as possible.
3. The services should be as secure as possible without affecting usability.
4. Enable the banking institutions to implement these services in a consistent manner.
5. An attempt should be made not to limit services to a specific transfer protocol if possible.

Design decisions

In light of the design goals a decision was made to implement the services as SOAP Web Services, accessible over the Internet through the HTTP secured with SSL (HTTPS) and Web Service Security (WSS) using the Username token and the X509 certificate token profiles.

The Internet is the most common and most inexpensive method of communicating between separate networks and HTTPS has the benefit of being a very common transfer protocol, accessible through most firewalls on port 443 (design goal 1).
The reason for choosing SSL instead of XML Encryption as defined in WS Security is that with regards to the current state of toolkits available to many programmers, the former is probably easier to implement (design goal 2, 3 and 4).
The use of certificates is deemed necessary, despite increased complexity for client implementations, as the level of security measures acceptable to the banking institutions and their clients rules out using username and passwords as the only barrier to accessing sensitive information (design goal 3).

SOAP is a standard for exchanging XML-based messages over a computer network. XML has various benefits as a message exchange format in B2B and B2C scenarios. Knowledge among programmers is widespread, a variety of tools for manipulation is available along with the fact that XML is human readable (design goal 2 and 3).
WSS has been a stable OASIS standard since 2004 and the UsernameToken profile is a relatively straightforward way of sending user authentication information in the context of a message. As the transport is secure and to simplify usage and server side implementation, the password is sent as clear text. Under WSS messages can also be signed in accordance with the XML Signature specification. Tools are available to most platforms that ease the use of WSS, such as WSE 2.0 and 3.0 for Microsoft .NET and the XWS-Security Framework for Java to name two (design goal 2, 3 and 4).

The reason that custom services and schemas were chosen was that no existing standard for financial services would have fit into the Icelandic banking environment without considerable modification. Most implement custom security models, some require specialized tools for usage and the learning curve would have been steep. Using XML schemas and the Web Service Description Language (WSDL) enables automatic tools to bootstrap the code for communicating with the services in a few minutes. The aim was that such code generation would create classes that would be as easy as possible to use, always keeping in mind that XML and object oriented structures do not always map directly (design goals 2).
The services offered are modelled on existing services and functionality that the banking institutions are all able to offer (design goal 5).

Finally it is worth mentioning that although this version of the services only offers HTTP as the application level protocol, it should be possible in the future to use other protocols (design goal 6), should that be deemed necessary.

Implementation

Overview

This following section describes some general issues that affect the use of the services and some code examples are given.

Behaviour

The following is a high level diagram of how a client will communicate with a server which hosts the web service, which then communicates with its back office systems.

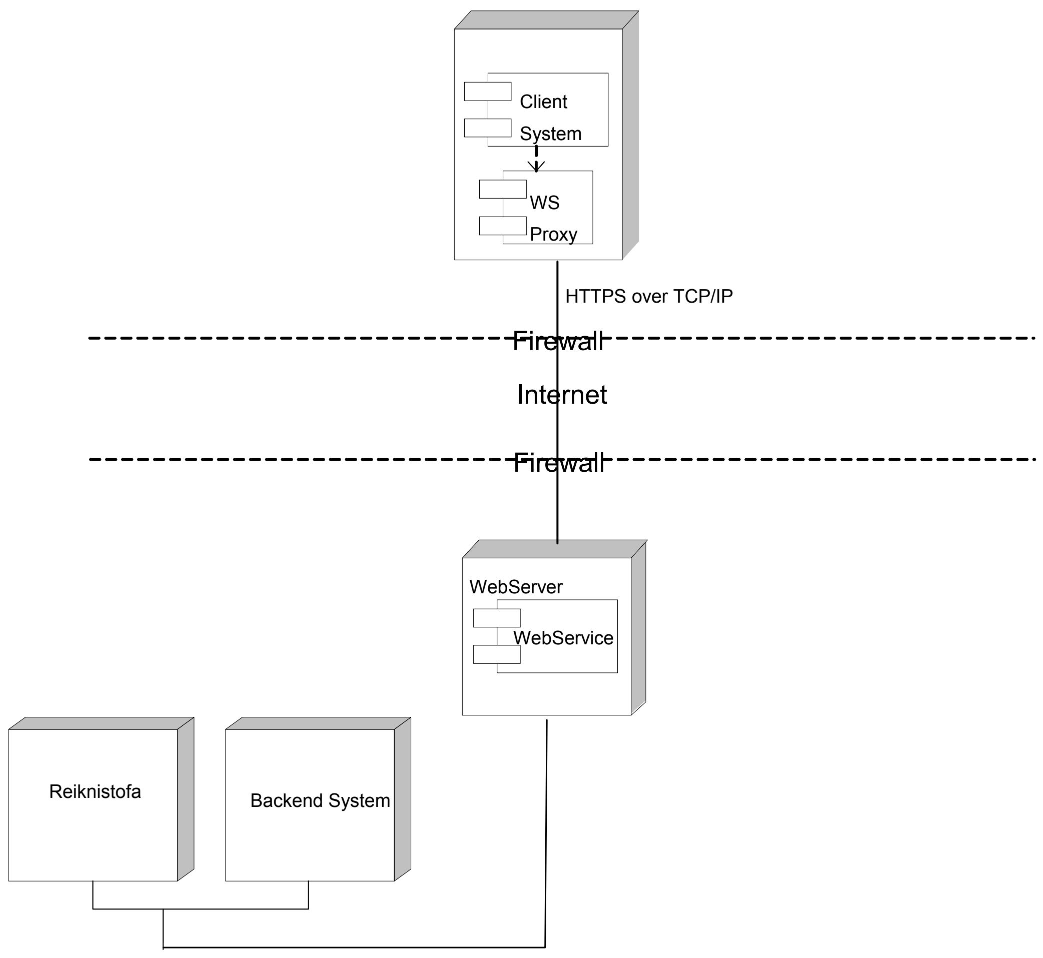


Figure 1. Basic setup.

Operations

The system tries to keep responses to operations are kept as quick and light as possible. It was decided that in most cases a user only wants to know that the operation has succeeded and does not need to receive the data he sent with the response. There are special operations designed to return the results of operations and all related data.

Exceptions

All communication is prone to exceptions and SOAP has a standard mechanism to communicate exceptions. These SOAP Exceptions are only thrown when it is not possible to complete an operation, usually due to faulty data or other technical reasons. Additionally, some circumstances where the input data does not conform to a given criteria can lead to an exception being thrown. When executing a batch, where it is possible for some operations to succeed but not others, other ways of returning error information is preferred.
Special error messages are returned in the details node of a SOAP exception when the error does not deal with SOAP headers. The different nodes returned in the details node are described in the following table:

Code	Type	Details
GeneralErrorCode	xs:string	Common error code across banking institutions.
GeneralErrorText	xs:string	Text to describe the GeneralErrorCode. ex: "Authentication failed", "Data could not be validated" etc.
BanksErrorCode	xs:string	Error code specific to the banking institution and the error instance.
BanksErrorText	xs:string	Text to describe the BanksErrorCode and/or data to resolve or help troubleshoot problems between banking institutions.

The BanksErrorCode can be used by each individual institution to identify individual error occurrences, e.g. to enable tracking. The GeneralErrorCodes are common error codes and indicate which class of error has occurred.

Code	Text	Details
0001	Service is Unavailable.	Implies that the service is closed for some reason.
1000	An error occurred.	A general error if a more detailed description is not available.
1100	Access to the operation is not present.	
1200	Data could not be validated.	The data could not be validated according to the XML schema.
1300	Business logic error.	Business rules were broken, e.g. dates or amounts were not valid.

All documents sent to the service are validated according to schema. Figure 1 shows a sample of an error.

```
<soap:Fault>
  <faultcode>soap:Client</faultcode>
  <faultstring>System.Web.Services.Protocols.SoapException: Payor ID could not
be validated at
KBBanki.Krofulina.WebServices.BankingClaims.CreateClaim(Claim
claim)</faultstring>
<faultactor>http://www.somewebserver.is/WebServices/2005/12/01/Claims.asmx</f
aultactor>
  <detail>
    <GeneralErrorCode>1200</GeneralErrorCode>
    <GeneralErrorText>Data could not be validated.</GeneralErrorText>
    <BanksErrorCode>5e80f0c6-479f-4dc2-9ab1-f2abed1e9f71</BanksErrorCode>
    <BanksErrorText>The 'Identifier' element has an invalid value according to its
data type. An error occurred at , (5, 78). BanksErrorText>
  </detail>
</soap:Fault>
```

Figure 1. A sample of a SOAP exception.

Timestamps

It is necessary to make sure that the clocks are as synchronized as possible on clients and servers. The reason for this is that SOAP messages include a "Time to live", which is important because the system will not perform operations which do not arrive within a reasonable time. The services in this document use a default time of 900 seconds.

UserNameToken

Each call to the service should include a UserNameToken in accordance with the OASIS WSS UsernameToken Profile 1.0. The token should include the Username and Password tags. The Password@Type attribute references by default the URI „...#PasswordText“ and the password should be sent as clear text.

```
<S11:Envelope xmlns:S11="..." xmlns:wsse="...">
  <S11:Header>
  ...
    <wsse:Security>
      <wsse:UsernameToken>
        <wsse:Username>MyUserName</wsse:Username>
        <wsse:Password>My1ongA$ndDlff9ltP%$$phr$se</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  ...
  </S11:Header>
...
</S11:Envelope>
```

Figure 2. A sample of a security header.

The Nonce and Created tags are optional and their usage will not be enforced server side.

Signing Messages

Digital signature of messages is mandatory. Each banking institution defines its own rules for which types of certificates can be used for the services. This means that certificate which is used at one bank may, or may not, be accepted by other banks.

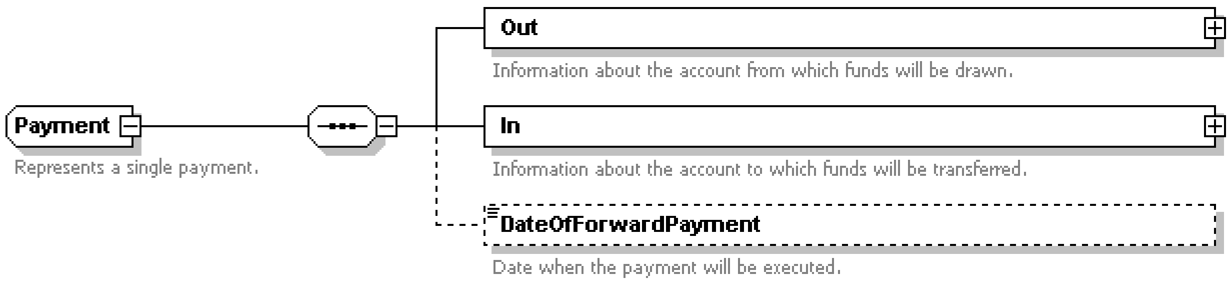
Icelandic Online

This section contains a manual for users of the web services for the banking institutions. The services described here are valid with all the banks, i.e. the same schemas and objects apply with all the banks.
The following paper describes the operations that can be performed in the first version of this standard. The operations are described in a manner of the way they are performed, i.e. that each operation is described in a way that depicts all the factors that need to be taken into consideration while performing each operation.
Pictures are used to further explain how objects are connected within each operation. Solid lines in these pictures indicate that the element in question must be entered, but the dotted lines indicate that the element is optional.
The schemas themselves will be accessible through other means of publication.

IcelandicOnlinePayments

Payment (DoPayment)

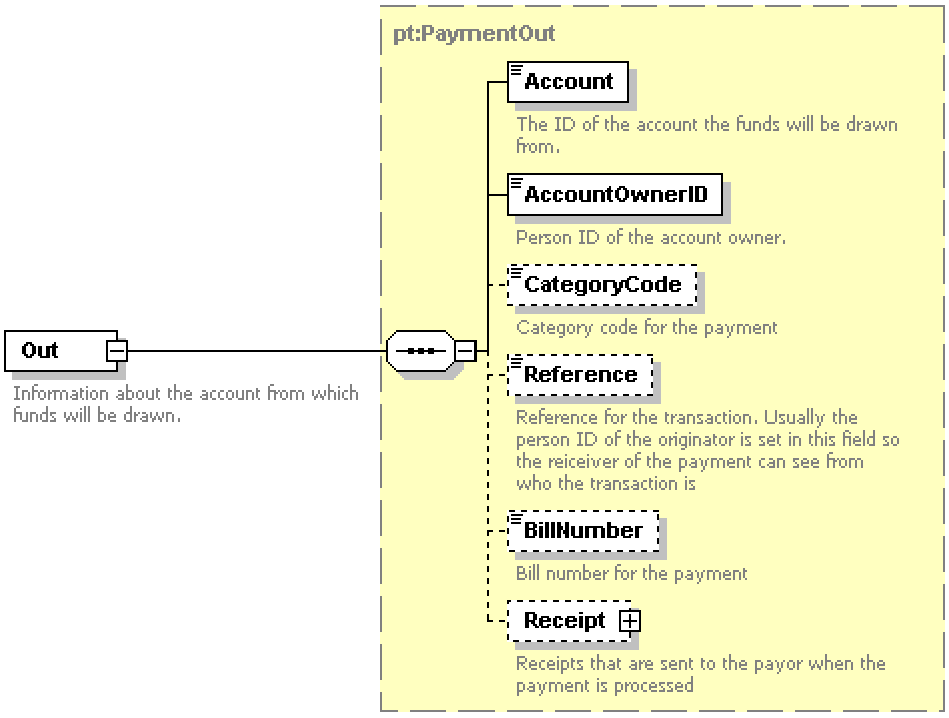
A description of how single payments are created. The object Payment has a list of PaymentOut and PaymentIn, which are the withdrawals and deposits, along with the date the payment shall be made. If a payment date is not entered, it is generally assumed that the payment shall be performed the very same day.


Payment
Represents a single payment.
Out
Information about the account from which funds will be drawn.
In
Information about the account to which funds will be transferred.
DateOfForwardPayment
Date when the payment will be executed.

A more detailed description of the sub-items of **Payment** follows:

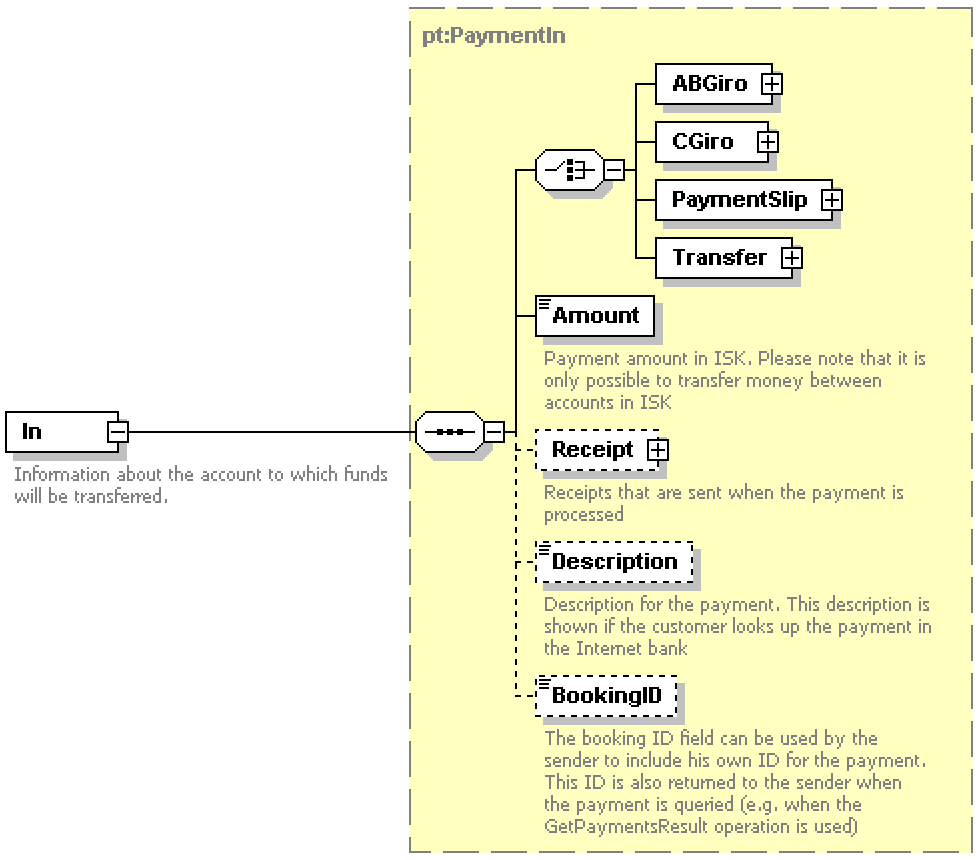
Out

Here we describe the withdrawal that takes place during payment. The only thing that must be entered here is the account number and the ID of the account owner. Category code, reference number and bill number can be entered and that information will be accessible when account statements are viewed. A receipt (intented for the payor) can also be sent.


pt:PaymentOut
Out
Information about the account from which funds will be drawn.
Account
The ID of the account the funds will be drawn from.
AccountOwnerID
Person ID of the account owner.
CategoryCode
Category code for the payment
Reference
Reference for the transaction. Usually the person ID of the originator is set in this field so the reiceiver of the payment can see from who the transaction is
BillNumber
Bill number for the payment
Receipt
Receipts that are sent to the payor when the payment is processed

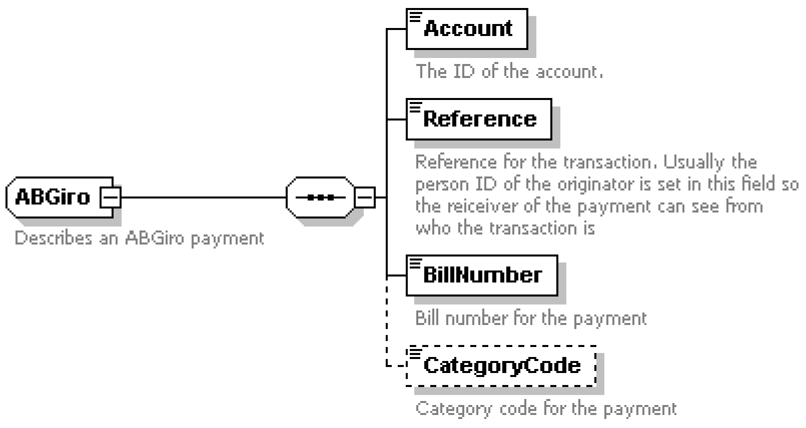
In

A choice is made between 4 types of deposits, AB giro, C giro, Payment bills and standard transfers. One of these must be selected. The amount in question must also be entered, but a receipt and a description of the payment are optional. The BookingId is thought as an supplementary field that the users can use to link payments into their own accounting systems.


pt:PaymentIn
ABGiro
CGiro
PaymentSlip
Transfer
Amount
Payment amount in ISK. Please note that it is only possible to transfer money between accounts in ISK
In
Information about the account to which funds will be transferred.
Receipt
Receipts that are sent when the payment is processed
Description
Description for the payment. This description is shown if the customer looks up the payment in the Internet bank
BookingID
The booking ID field can be used by the sender to include his own ID for the payment. This ID is also returned to the sender when the payment is queried (e.g. when the GetPaymentsResult operation is used)

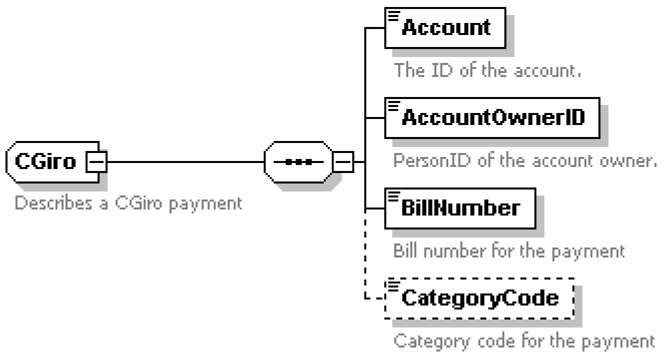
ABGiro

Obligatory fields are the account ID (to which money will be deposited), the reference number for the payment and the bill number of the giro to be paid. An optional field is also available for the category code.


ABGiro
Describes an ABGiro payment
Account
The ID of the account.
Reference
Reference for the transaction. Usually the person ID of the originator is set in this field so the reiceiver of the payment can see from who the transaction is
BillNumber
Bill number for the payment
CategoryCode
Category code for the payment

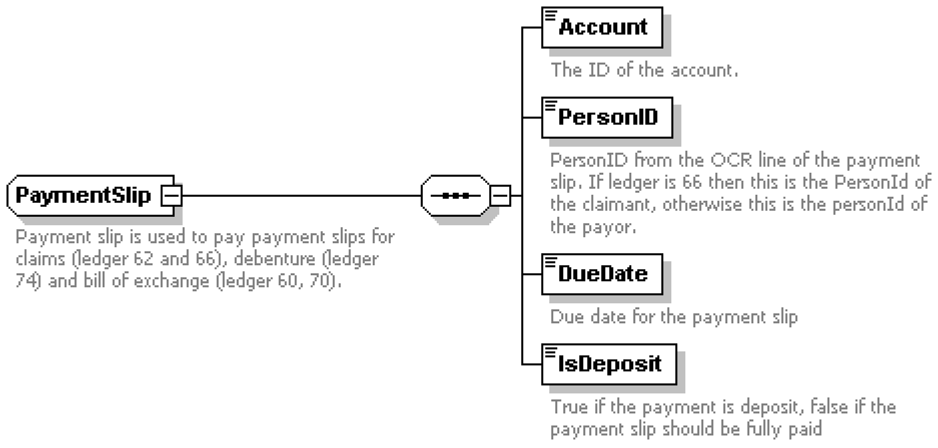
CGiro

Obligatory fields are the account ID (to which money will be deposited), the personal ID of the account owner and the bill number for the giro. The category code field is optional.


CGiro
Describes a CGiro payment
Account
The ID of the account.
AccountOwnerID
PersonID of the account owner.
BillNumber
Bill number for the payment
CategoryCode
Category code for the payment

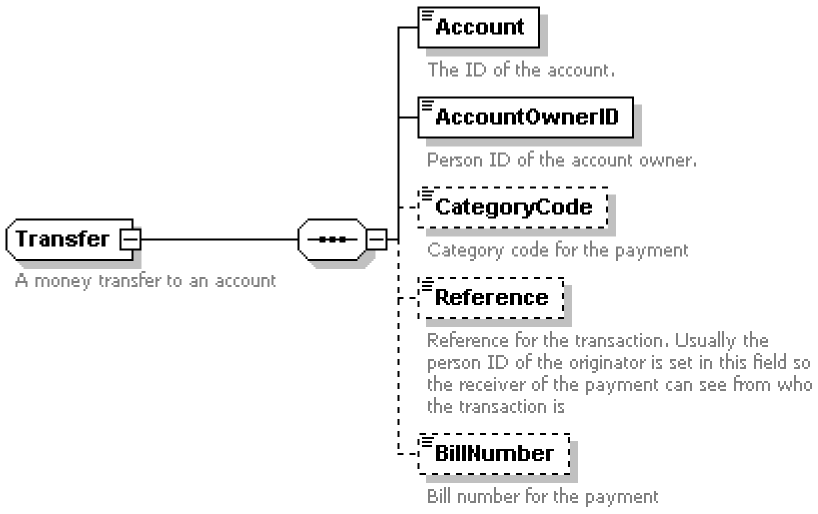
PaymentSlip

All fields are obligatory, the account (to which funds will be deposited), the ID number of the payor og invoicer (depends on the ledger), the due date of the slip and the IsDeposit fields dictates whether this payment is a partial or complete payment of the slip.


PaymentSlip
Payment slip is used to pay payment slips for claims (ledger 62 and 66), debenture (ledger 74) and bill of exchange (ledger 60, 70).
Account
The ID of the account.
PersonID
PersonID from the OCR line of the payment slip. If ledger is 66 then this is the PersonId of the claimant, otherwise this is the personId of the payor.
DueDate
Due date for the payment slip
IsDeposit
True if the payment is deposit, false if the payment slip should be fully paid

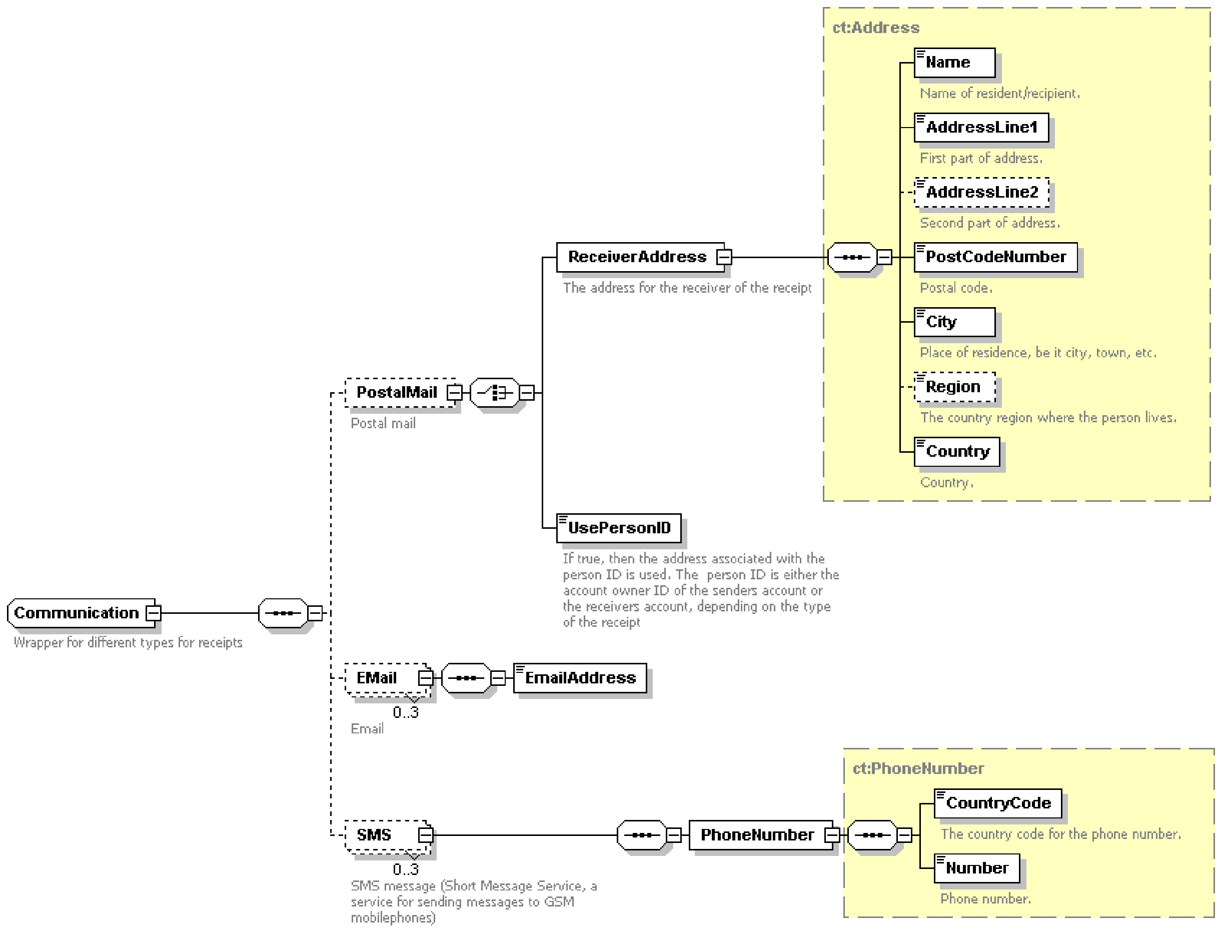
Transfer

A standard transfer to an account. Obligatory are the account number and the ID number of the account owner fields (of the account that funds will be deposited to), and optional fields are for the category code, reference number and the bill number.


Transfer
A money transfer to an account
Account
The ID of the account.
AccountOwnerID
Person ID of the account owner.
CategoryCode
Category code for the payment
Reference
Reference for the transaction. Usually the person ID of the originator is set in this field so the receiver of the payment can see from who the transaction is
BillNumber
Bill number for the payment

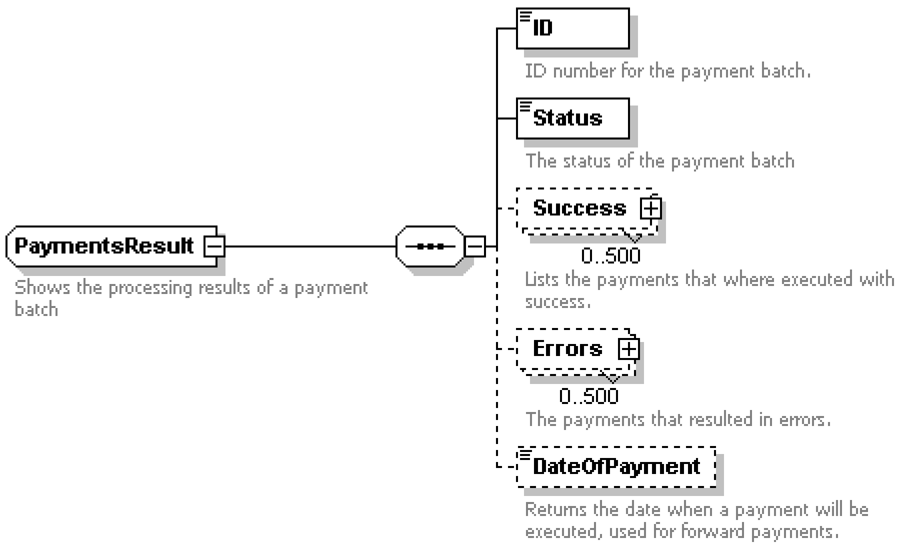
Receipt

Receipts are sent to the payment recipient when it is performed. This is an optional field on both in and out payments. The choice stands between sending PostalMail (a standard letter mail), Email and SMS. The options are sending 1 PostalMail, 3 emails and 3 sms's. If PostalMail is selected, then a recipient must either be entered by using the ReceiverAddress or by setting the UsePersonID field as true, in which case a receipt is sent so the recipients home as listed in the national register. The ReceiverAddress consists of a name, two address lines, the postal code, city, region and country.
If it is selected to send an email, then only the email address must be entered.
If it is selected to send an SMS, then a country code and phone number must be entered.


ct:Address
Name
Name of resident/recipient.
AddressLine1
First part of address.
AddressLine2
Second part of address.
ReceiverAddress
The address for the receiver of the receipt
PostCodeNumber
Postal code.
City
Place of residence, be it city, town, etc.
PostalMail
Postal mail
Region
The country region where the person lives.
Country
Country.
UsePersonID
If true, then the address associated with the person ID is used. The person ID is either the account owner ID of the senders account or the receivers account, depending on the type of the receipt
Communication
Wrapper for different types for receipts
EMail
0..3
Email
EmailAddress
ct:PhoneNumber
CountryCode
The country code for the phone number.
SMS
PhoneNumber
0..3
SMS message (Short Message Service, a service for sending messages to GSM mobilephones)
Number
Phone number.

PaymentsResult (DoPaymentResponse)

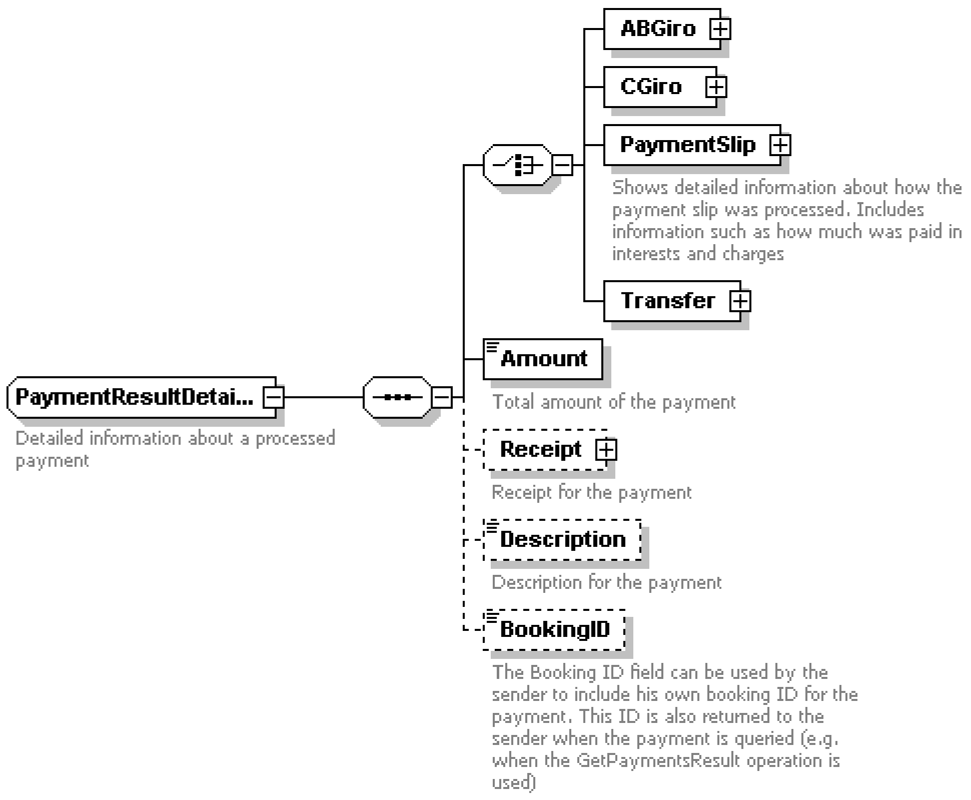
This is a description of the response to a creation of a single payment (the results for a payment batch). ID is the unique identification for the batch that was created and Status is the batch status. Success and Errors are kept optional, becaust it is always possible that one of those would be completely empty. In Success and Errors the payments just created are being returned. If a future date for payment was selected in the payment creation, then that date is returned here in DateOfPayment, and the status of the batch is put as OnHold.


ID
ID number for the payment batch.
Status
The status of the payment batch
Success
0..500
Lists the payments that where executed with success.
PaymentsResult
Shows the processing results of a payment batch
Errors
0..500
The payments that resulted in errors.
DateOfPayment
Returns the date when a payment will be executed, used for forward payments.

A further description of the things PaymentsResult consists of:

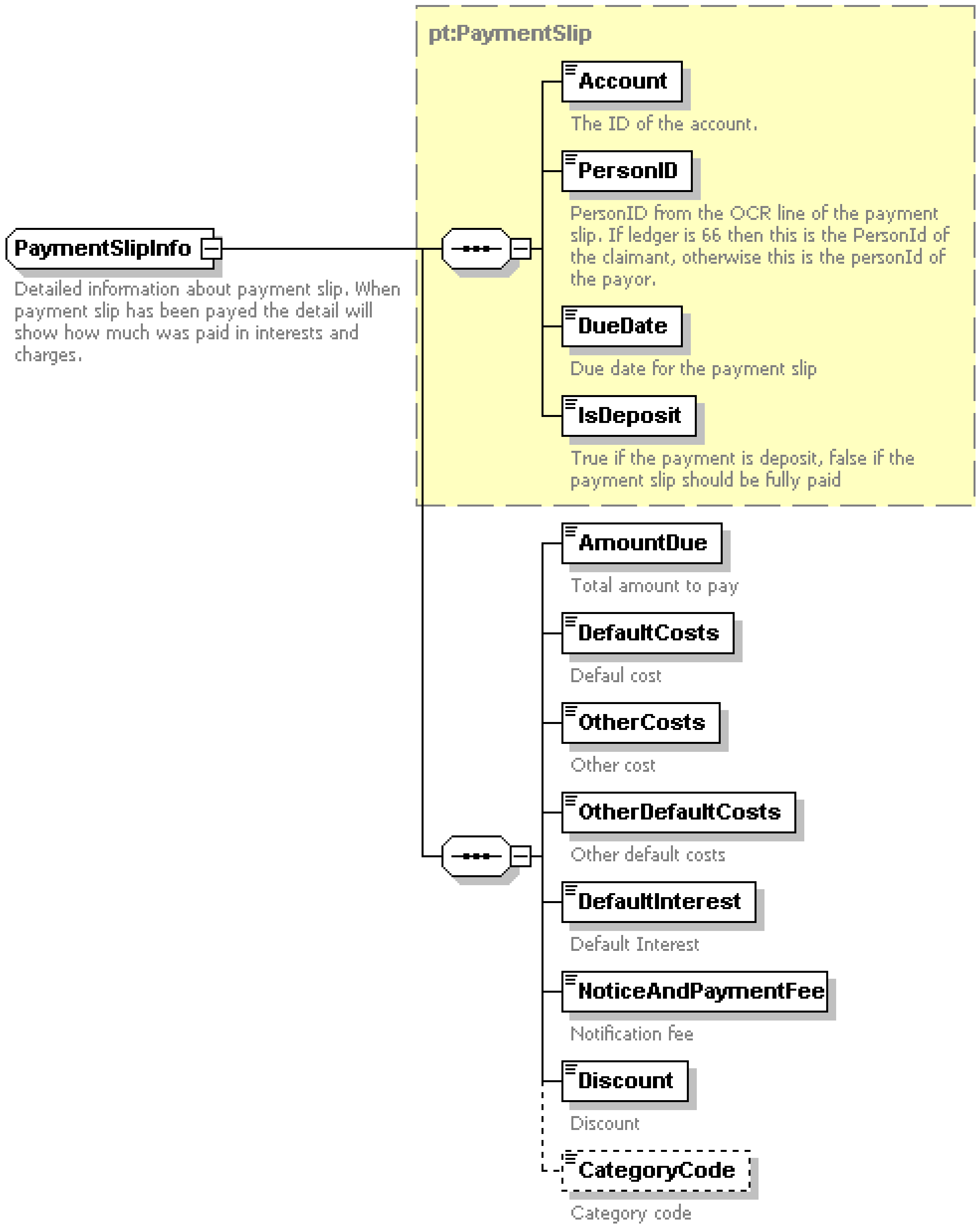
Success

A list of payments that were successfully performed. The amount of the payment is shown. ABGiro, CGiro and Transfer are identical to the actual payment, but PaymentSlip changes in the way that more detailed information about the interests and fees for the payment is given.


ABGiro
CGiro
PaymentSlip
Shows detailed information about how the payment slip was processed. Includes information such as how much was paid in interests and charges
Transfer
PaymentResultDetai...
Detailed information about a processed payment
Amount
Total amount of the payment
Receipt
Receipt for the payment
Description
Description for the payment
BookingID
The Booking ID field can be used by the sender to include his own booking ID for the payment. This ID is also returned to the sender when the payment is queried (e.g. when the GetPaymentsResult operation is used)

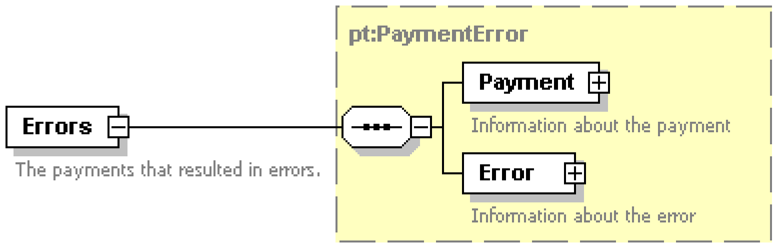
PaymentSlip (in Success)

The key in the PaymentSlip that is a part of the payment, is only a little part of the answer, as a part of PaymentSlipInfo. Added to it are details about the payment slip.


pt:PaymentSlip
PaymentSlipInfo
Detailed information about payment slip. When payment slip has been payed the detail will show how much was paid in interests and charges.
Account
The ID of the account.
PersonID
PersonID from the OCR line of the payment slip. If ledger is 66 then this is the PersonId of the claimant, otherwise this is the personId of the payor.
DueDate
Due date for the payment slip
IsDeposit
True if the payment is deposit, false if the payment slip should be fully paid
AmountDue
Total amount to pay
DefaultCosts
Defaul cost
OtherCosts
Other cost
OtherDefaultCosts
Other default costs
DefaultInterest
Default Interest
NoticeAndPaymentFee
Notification fee
Discount
Discount
CategoryCode
Category code

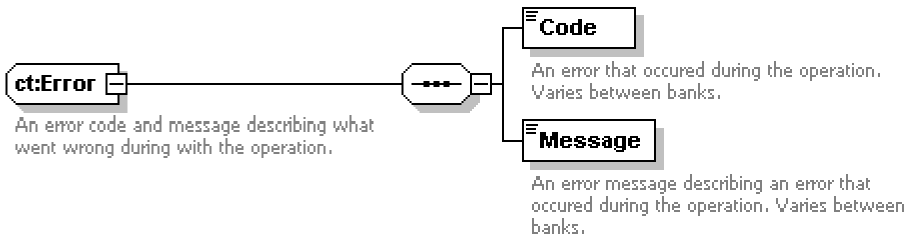
Errors

A list of the payments that an error occured on and could therefore not be created. The item Payment is identical to the one previously described in this document.


pt:PaymentError
Errors
The payments that resulted in errors.
Payment
Information about the payment
Error
Information about the error

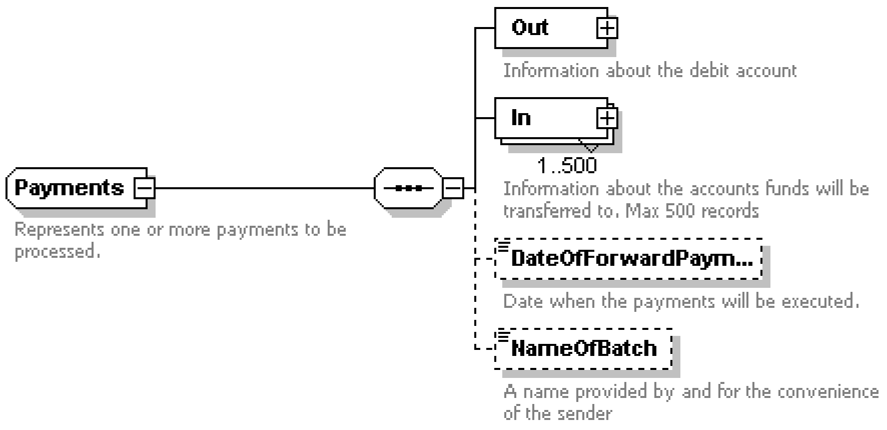
Error

A more detailed description of the error that occurred. Code is the number of the error and Message a description of the error that occurred.


ct:Error
An error code and message describing what went wrong during with the operation.
Code
An error that occured during the operation. Varies between banks.
Message
An error message describing an error that occured during the operation. Varies between banks.

Payments (DoPayments)

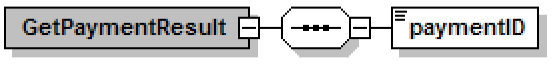
A description of how a list of payments is created. The element Payments has one field for a withdrawal and 1 to 500 possible deposits. Payments also has two attributes, RollbackOnError and IsOneToMany. RollbackOnError means that if any one of the payments fails, then all payments are cancelled. IsOneToMany indicates whether one withdrawal should be made for the entire batch or if one withdrawal should be made per deposit. A date for forward payment and a batch name can also be entered, but those elements are optional. In and Out elements are identical to the ones in creation of a single payment


Out
Information about the debit account
In
1..500
Information about the accounts funds will be transferred to. Max 500 records
Payments
Represents one or more payments to be processed.
DateOfForwardPaym...
Date when the payments will be executed.
NameOfBatch
A name provided by and for the convenience of the sender

The response to a batch creation is OperationID which is a string variable that is an identifier for the operation.

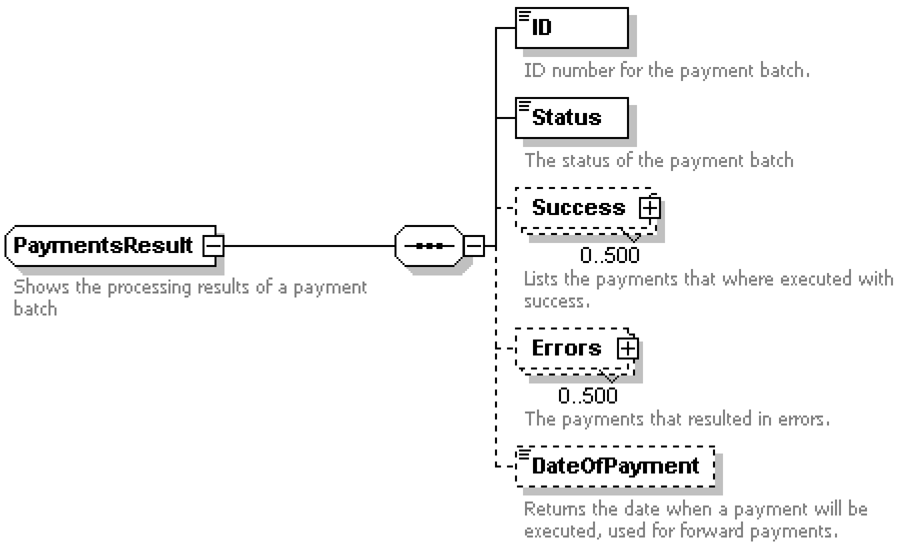
Payment query (GetPaymentResult)

An OperationID (string) is sent for the payment to be fetched.



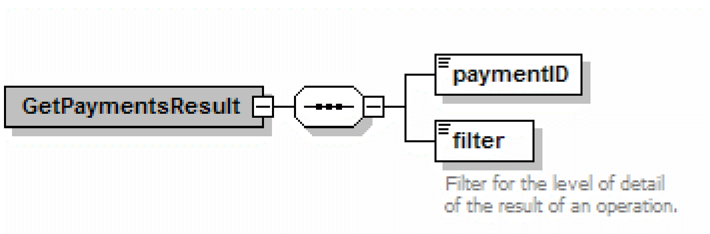
PaymentsResult

The response to the query. Same answer as to the creation of a single payment.


ID
ID number for the payment batch.
Status
The status of the payment batch
Success
0..500
Lists the payments that where executed with success.
PaymentsResult
Shows the processing results of a payment batch
Errors
0..500
The payments that resulted in errors.
DateOfPayment
Returns the date when a payment will be executed, used for forward payments.

Payments query (GetPaymentsResult)

A query is sent that consists of a paymentID (string), and a filder that contains PaymentStatus. Using the filter, it is possible to get the status of payments (GetStatus), get all payments on errors (GetErrors), get all successful payments (GetOkay) and getting all payments (GetAll).

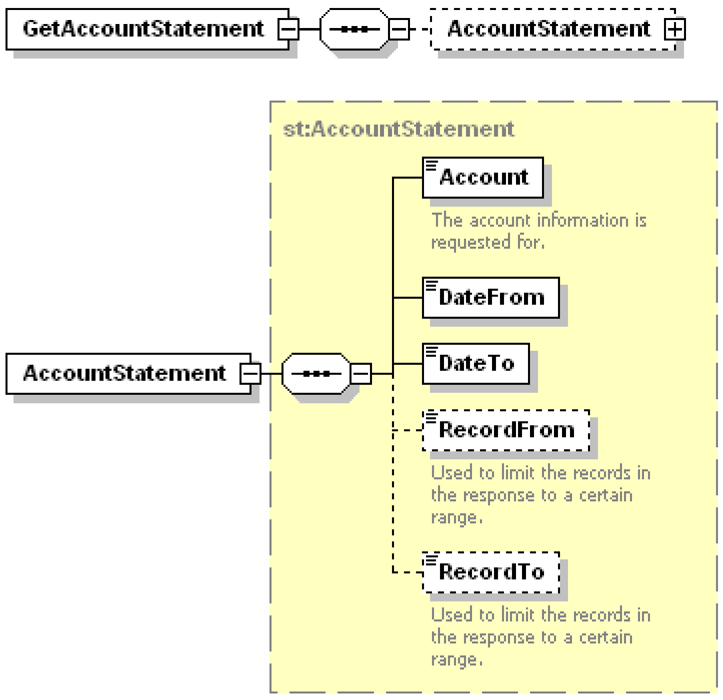

GetPaymentsResult
paymentID
filter
Filter for the level of detail of the result of an operation.

The answer to this query is the same as in GetPaymentResult, except that in this case it is a lot more likely that the lists are used more than in the single payment.

IcelandicOnlineStatements

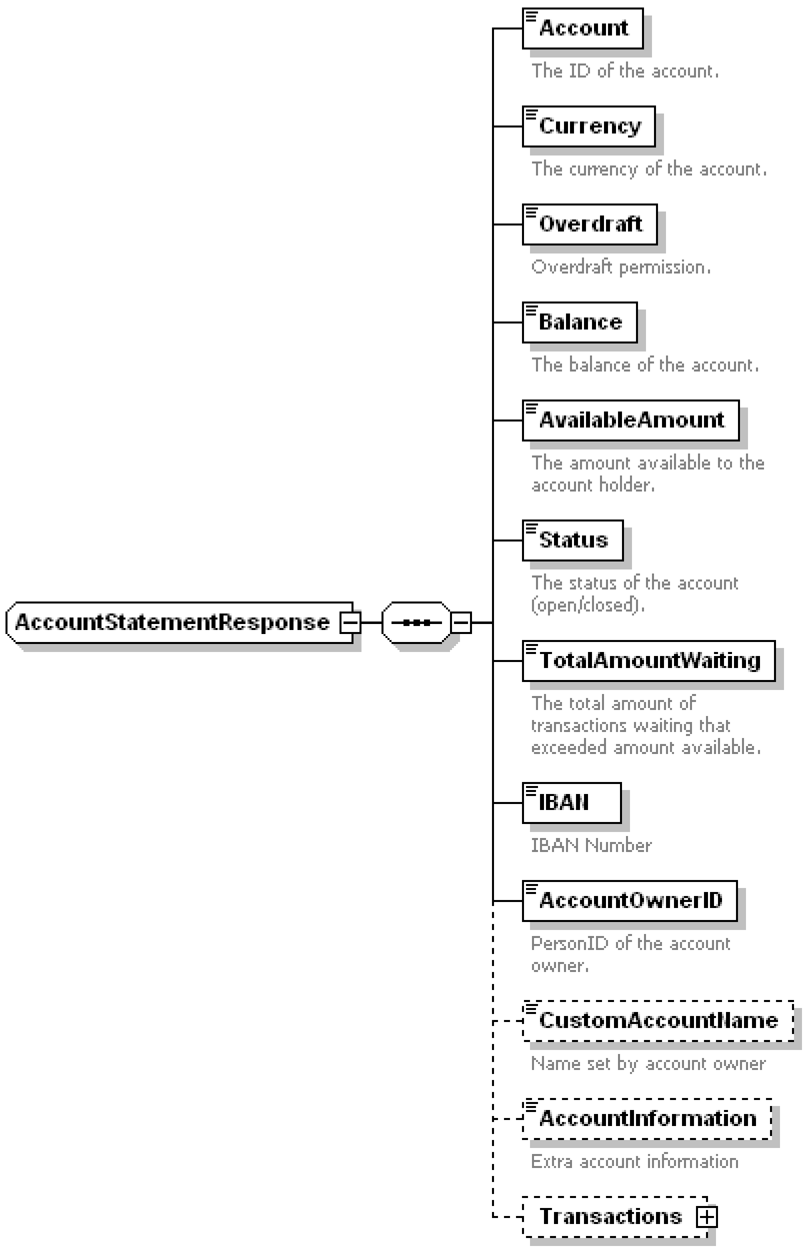
AccountStatement (GetAccountStatement):

GetAccountStatement has one element AccountStatement, which is used to perform a query on an account. The obligatory fields for this query are the account number (Account) and the start and end dates of the statement. It is also possible to select specific records from within the statement. This is added for the user, in case there are very many entries within the same period.


GetAccountStatement
AccountStatement
st:AccountStatement
Account
The account information is requested for.
DateFrom
DateTo
RecordFrom
Used to limit the records in the response to a certain range.
RecordTo
Used to limit the records in the response to a certain range.
AccountStatement

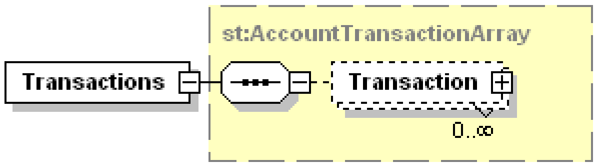
Get AccountStatementResponse

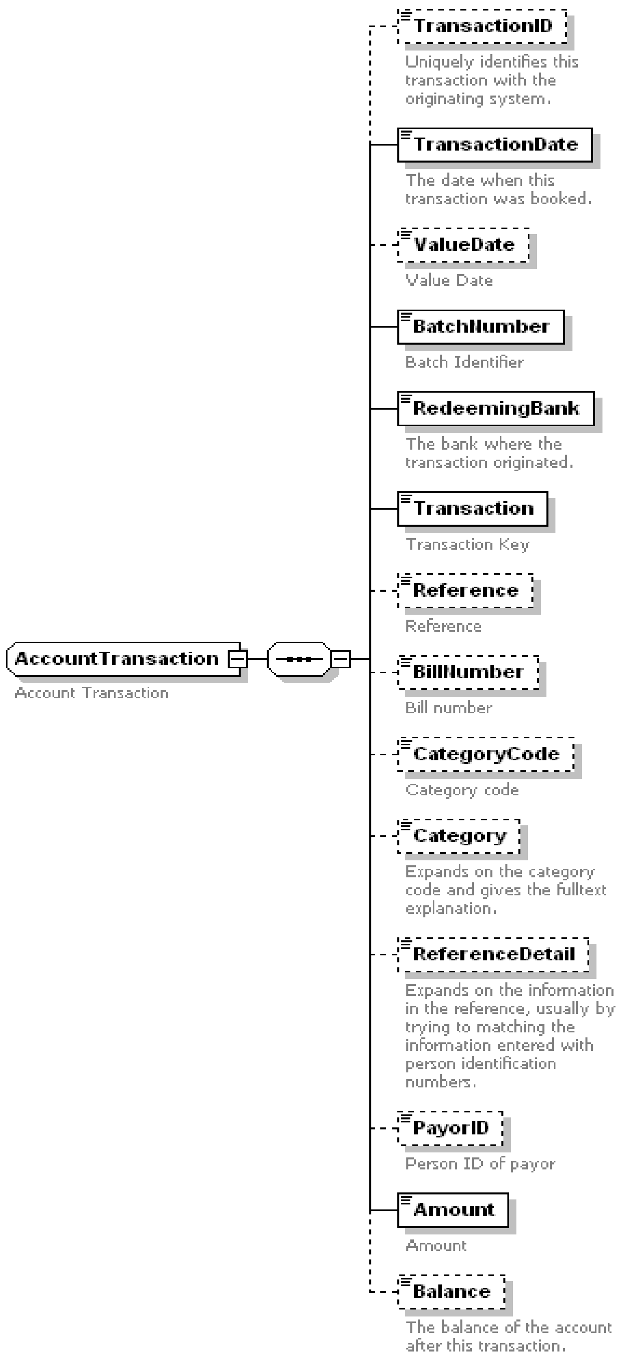
An account statement, contains information about the account itself, as well as allt the account entries (Transactions), but that element is not returned if no entries were found.


AccountStatementResponse
Account
The ID of the account.
Currency
The currency of the account.
Overdraft
Overdraft permission.
Balance
The balance of the account.
AvailableAmount
The amount available to the account holder.
Status
The status of the account (open/closed).
TotalAmountWaiting
The total amount of transactions waiting that exceeded amount available.
IBAN
IBAN Number
AccountOwnerID
PersonID of the account owner.
CustomAccountName
Name set by account owner
AccountInformation
Extra account information
Transactions

Transactions

The element Transaction in the account statement contains a list of entries (AccountTransactionArray). Each account transaction has a detailed description of the information regarding an account statement entry.

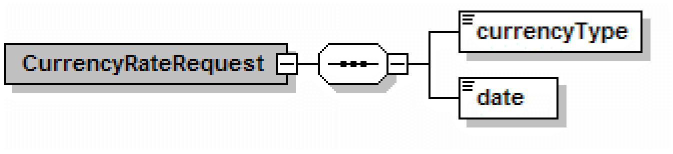




TransactionID
Uniquely identifies this transaction with the originating system.
TransactionDate
The date when this transaction was booked.
ValueDate
Value Date
BatchNumber
Batch Identifier
RedeemingBank
The bank where the transaction originated.
Transaction
Transaction Key
Reference
Reference
AccountTransaction
Account Transaction
BillNumber
Bill number
CategoryCode
Category code
Category
Expands on the category code and gives the fulltext explanation.
ReferenceDetail
Expands on the information in the reference, usually by trying to matching the information entered with person identification numbers.
PayorID
Person ID of payor
Amount
Amount
Balance
The balance of the account after this transaction.

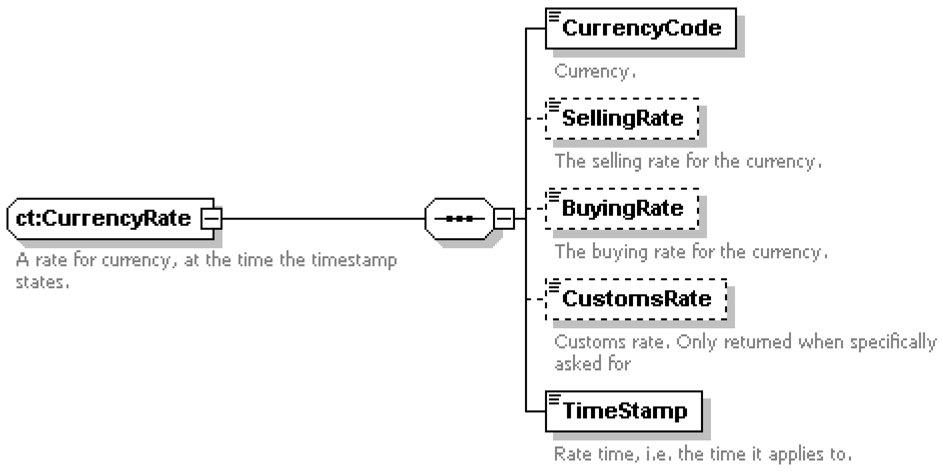
CurrencyRateRequest:

When a query is made regarding currency rate, the date of the rate in question is entered, as well as a CurrencyType element which dictates which type of rate is to be fetched.



Currency rate (CurrencyRateResponse):

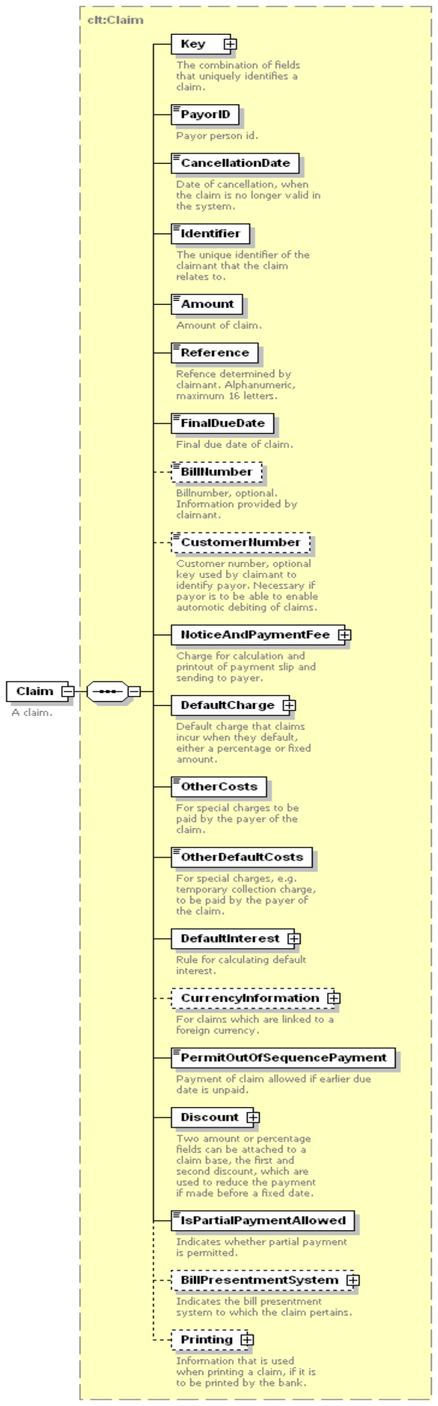
The response returns a list of CurrencyRate elements. The rate elements are made optional because if the query is made for customs rate, then only the customs rate is returned, and not the selling rate or buying rate. It's the same thing when the query is made for note rate or exchange rate.


ct:CurrencyRate
A rate for currency, at the time the timestamp states.
CurrencyCode
Currency.
SellingRate
The selling rate for the currency.
BuyingRate
The buying rate for the currency.
CustomsRate
Customs rate. Only returned when specifically asked for
TimeStamp
Rate time, i.e. the time it applies to.

IcelandicOnlineClaims

Claim creation/Claim modification(CreateClaims/AlterClaims)

Receives a list claims, that consists of Claim elements.



Key: The combination of fields that uniquely identifies a claim.

PayorID: Payor person id.

CancellationDate: Date of cancellation, when the claim is no longer valid in the system.

Identifier: The unique identifier of the claimant that the claim relates to.

Amount: Amount of claim.

Reference: Refence determined by claimant. Alphanumeric, maximum 16 letters.

FinalDueDate: Final due date of claim.

BillNumber: Billnumber, optional. Information provided by claimant.

CustomerNumber: Customer number, optional key used by claimant to identify payor. Necessary if payor is to be able to enable automotic debiting of claims.

NoticeAndPaymentFee: Charge for calculation and printout of payment slip and sending to payer.

DefaultCharge: Default charge that claims incur when they default, either a percentage or fixed amount.

OtherCosts: For special charges to be paid by the payer of the claim.

OtherDefaultCosts: For special charges, e.g. temporary collection charge, to be paid by the payer of the claim.

DefaultInterest: Rule for calculating default interest.

CurrencyInformation: For claims which are linked to a foreign currency.

PermitOutOfSequencePayment: Payment of claim allowed if earlier due date is unpaid.

Discount: Two amount or percentage fields can be attached to a claim base, the first and second discount, which are used to reduce the payment if made before a fixed date.

IsPartialPaymentAllowed: Indicates whether partial payment is permitted.

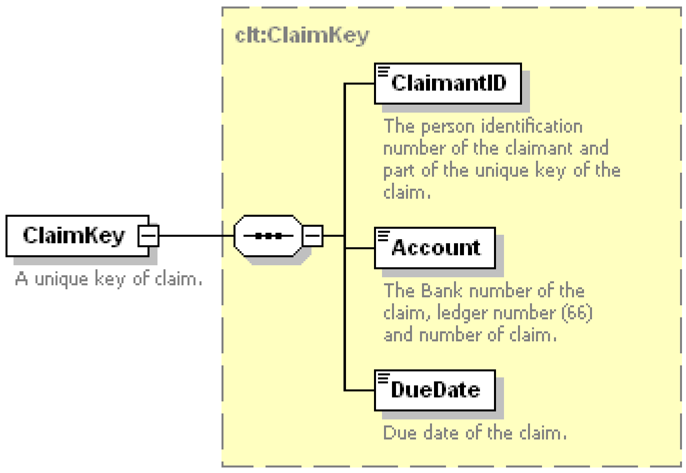
BillPresentmentSystem: Indicates the bill presentment system to which the claim pertains.

Printing: Information that is used when printing a claim, if it is to be printed by the bank.

A more detailed description of the elements that a claim consists of:

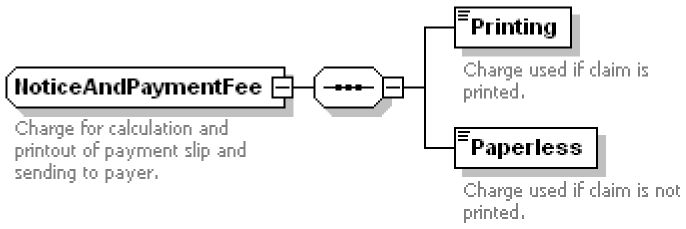
ClaimKey

A unique key for a claim, that consist of the personal ID of the claimant, the bank number of the claim and it's due date.


clt:ClaimKey
ClaimKey
A unique key of claim.
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

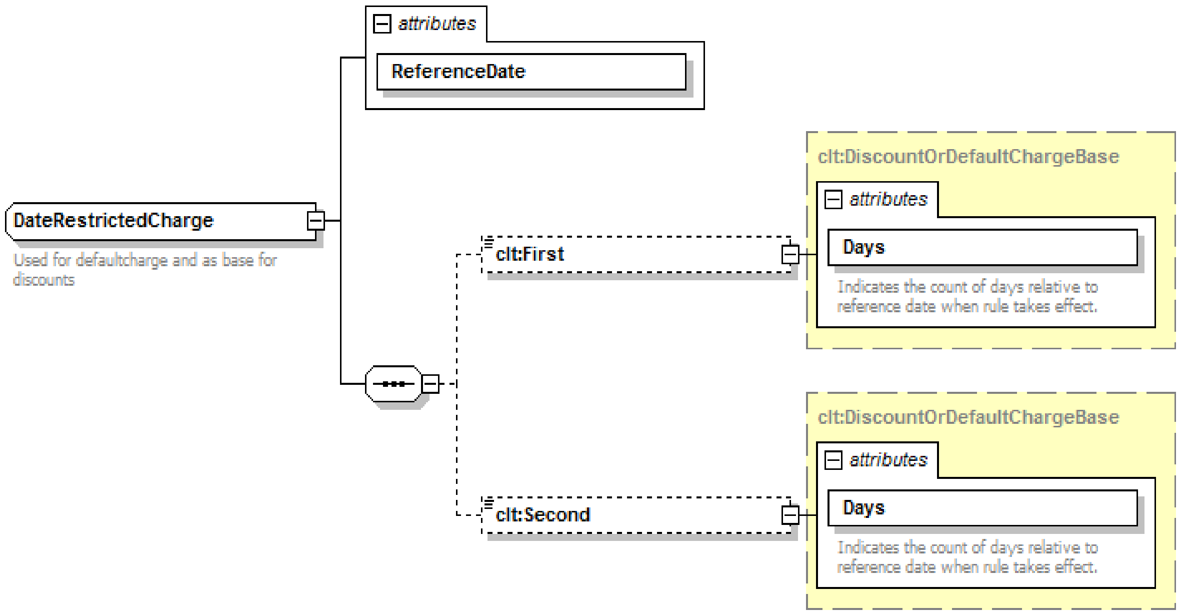
NoticeAndPaymentFee

The fee for sending a notice to the payor, but printing out the claim is optional.


NoticeAndPaymentFee
Charge for calculation and printout of payment slip and sending to payer.
Printing
Charge used if claim is printed.
Paperless
Charge used if claim is not printed.

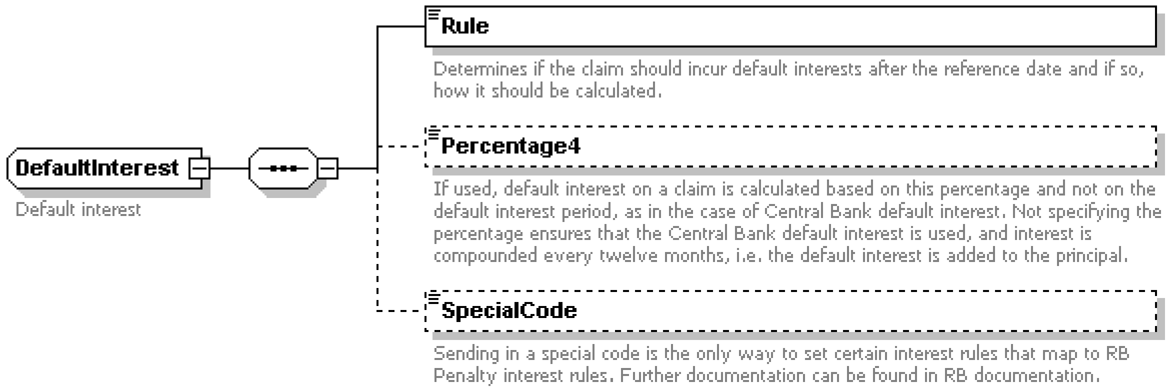
DefaultCharge

A charge that is added to claims once they end up in default. There are 2 different charges, first and second default charge. They both consist of an amount and a percentage.


attributes
ReferenceDate
DateRestrictedCharge
Used for defaultcharge and as base for discounts
clt:First
clt:DiscountOrDefaultChargeBase
attributes
Days
Indicates the count of days relative to reference date when rule takes effect.
clt:Second
clt:DiscountOrDefaultChargeBase
attributes
Days
Indicates the count of days relative to reference date when rule takes effect.

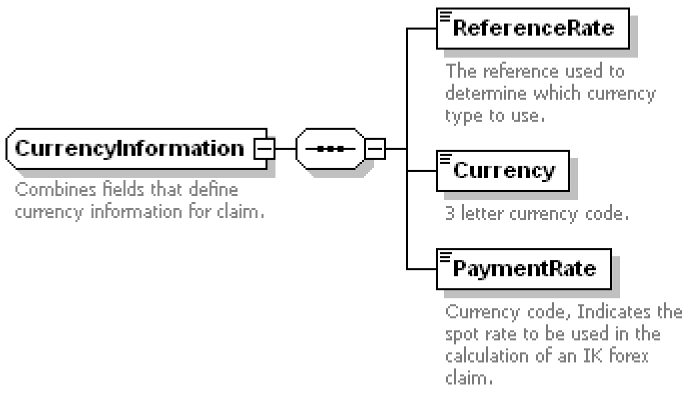
DefaultInterest

Dictates which default interest rule to use if a claim becomes default.


DefaultInterest
Default interest
Rule
Determines if the claim should incur default interests after the reference date and if so, how it should be calculated.
Percentage4
If used, default interest on a claim is calculated based on this percentage and not on the default interest period, as in the case of Central Bank default interest. Not specifying the percentage ensures that the Central Bank default interest is used, and interest is compounded every twelve months, i.e. the default interest is added to the principal.
SpecialCode
Sending in a special code is the only way to set certain interest rules that map to RB Penalty interest rules. Further documentation can be found in RB documentation.

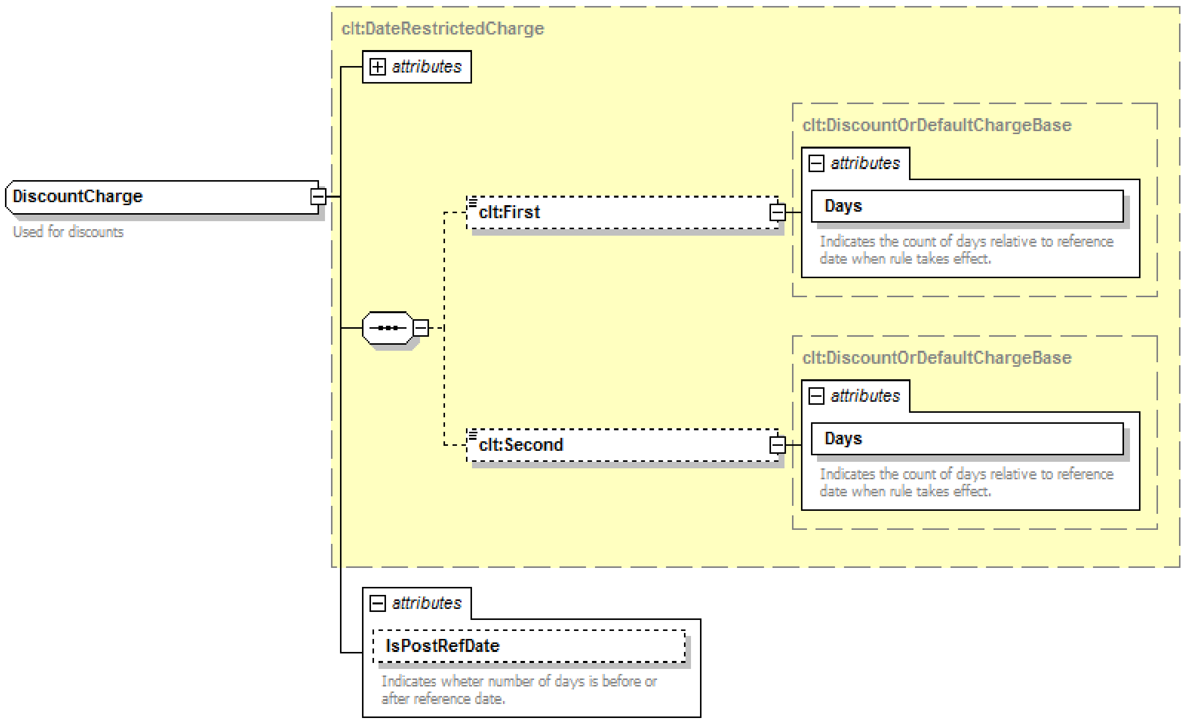
Currency information

An optional element on a claim, but is used for currency claims.


CurrencyInformation
Combines fields that define currency information for claim.
ReferenceRate
The reference used to determine which currency type to use.
Currency
3 letter currency code.
PaymentRate
Currency code, Indicates the spot rate to be used in the calculation of an IK forex claim.

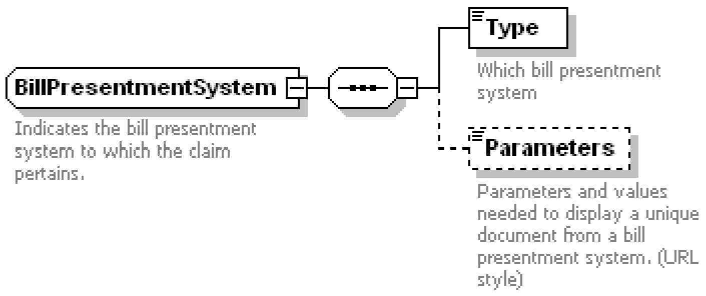
Discount

Which discounts are given on a claim, and look very similar to the default charge.


clt:DateRestrictedCharge
attributes
DiscountCharge
Used for discounts
clt:First
clt:DiscountOrDefaultChargeBase
attributes
Days
Indicates the count of days relative to reference date when rule takes effect.
clt:Second
clt:DiscountOrDefaultChargeBase
attributes
Days
Indicates the count of days relative to reference date when rule takes effect.
attributes
IsPostRefDate
Indicates wheter number of days is before or after reference date.

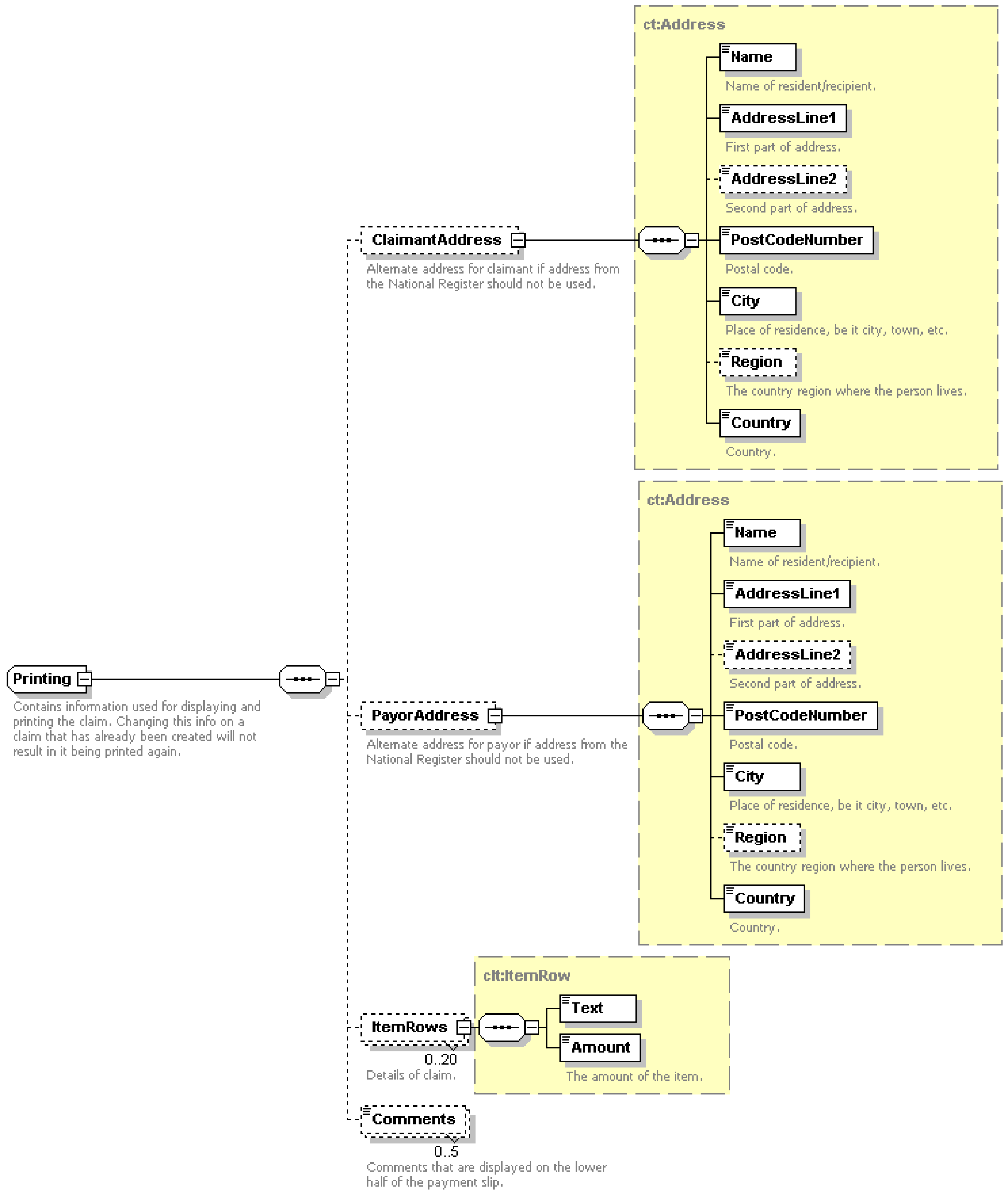
Bill Presentment System

Which presentment system to use, determined by the Type element, and a reference to specific system using parameters.


BillPresentmentSystem
Indicates the bill presentment system to which the claim pertains.
Type
Which bill presentment system
Parameters
Parameters and values needed to display a unique document from a bill presentment system. (URL style)

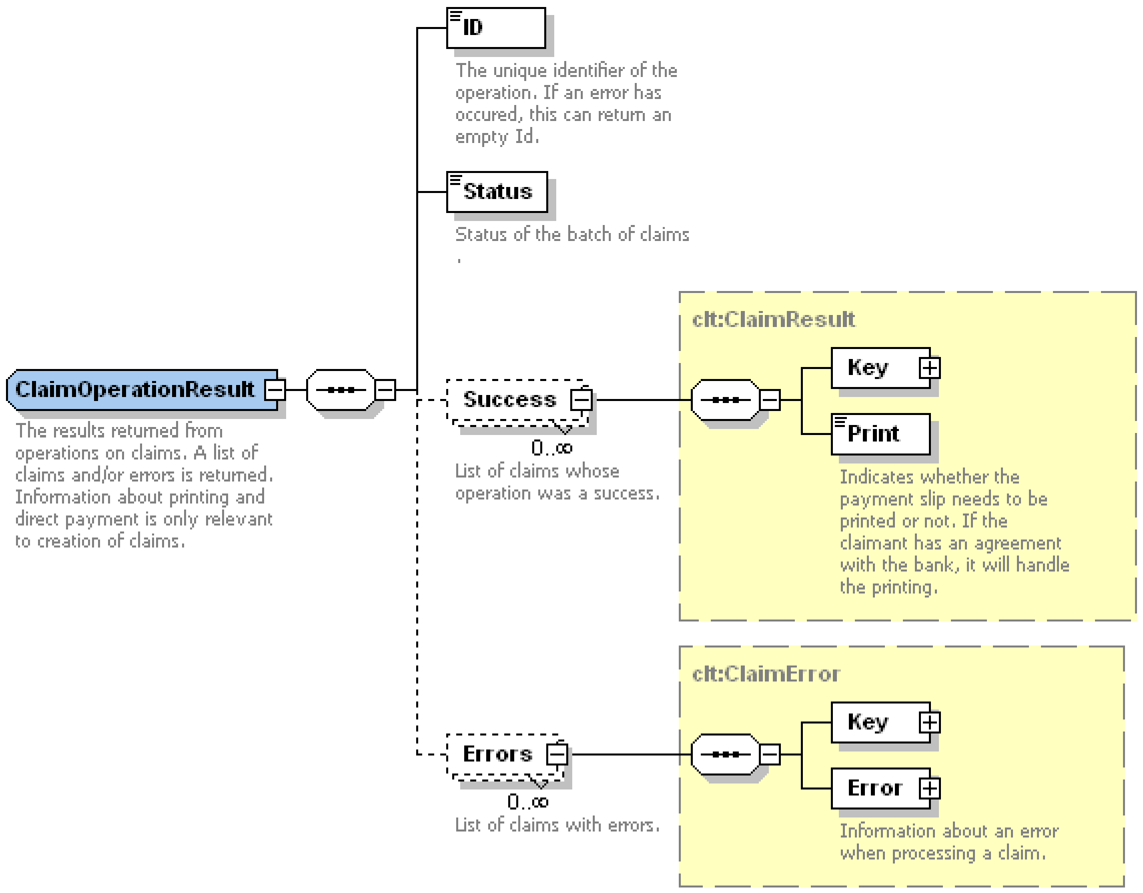
Printing

A description of how a claim is to be printed if done so by a banking institution. All elements in the printing section are optional.


ct:Address
Name
Name of resident/recipient.
AddressLine1
First part of address.
AddressLine2
Second part of address.
PostCodeNumber
Postal code.
City
Place of residence, be it city, town, etc.
Region
The country region where the person lives.
Country
Country.
ClaimantAddress
Alternate address for claimant if address from the National Register should not be used.
Printing
Contains information used for displaying and printing the claim. Changing this info on a claim that has already been created will not result in it being printed again.
PayorAddress
Alternate address for payor if address from the National Register should not be used.
ct:Address
Name
Name of resident/recipient.
AddressLine1
First part of address.
AddressLine2
Second part of address.
PostCodeNumber
Postal code.
City
Place of residence, be it city, town, etc.
Region
The country region where the person lives.
Country
Country.
clt:ItemRow
ItemRows
0..20
Details of claim.
Text
Amount
The amount of the item.
Comments
0..5
Comments that are displayed on the lower half of the payment slip.

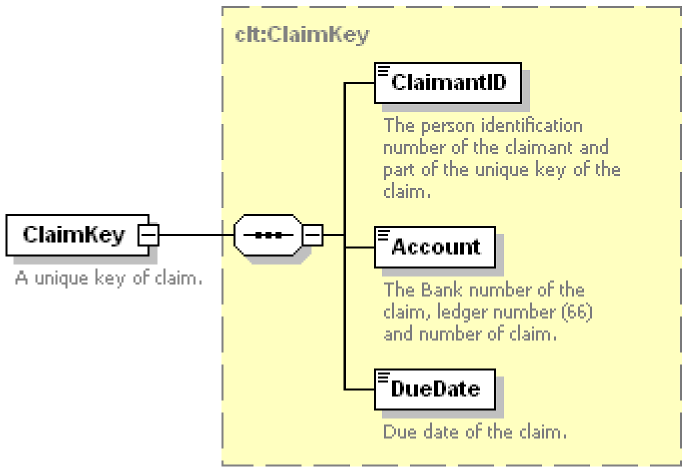
ClaimOperationResult

Information about the result of an operation. A list of claims and/or errrors is returned. Information about printing and direct payment only apply when a claim is created.


ID
The unique identifier of the operation. If an error has occured, this can return an empty Id.
Status
Status of the batch of claims .
ClaimOperationResult
The results returned from operations on claims. A list of claims and/or errors is returned. Information about printing and direct payment is only relevant to creation of claims.
Success
0..∞
List of claims whose operation was a success.
clt:ClaimResult
Key
Print
Indicates whether the payment slip needs to be printed or not. If the claimant has an agreement with the bank, it will handle the printing.
Errors
0..∞
List of claims with errors.
clt:ClaimError
Key
Error
Information about an error when processing a claim.

CancelClaims

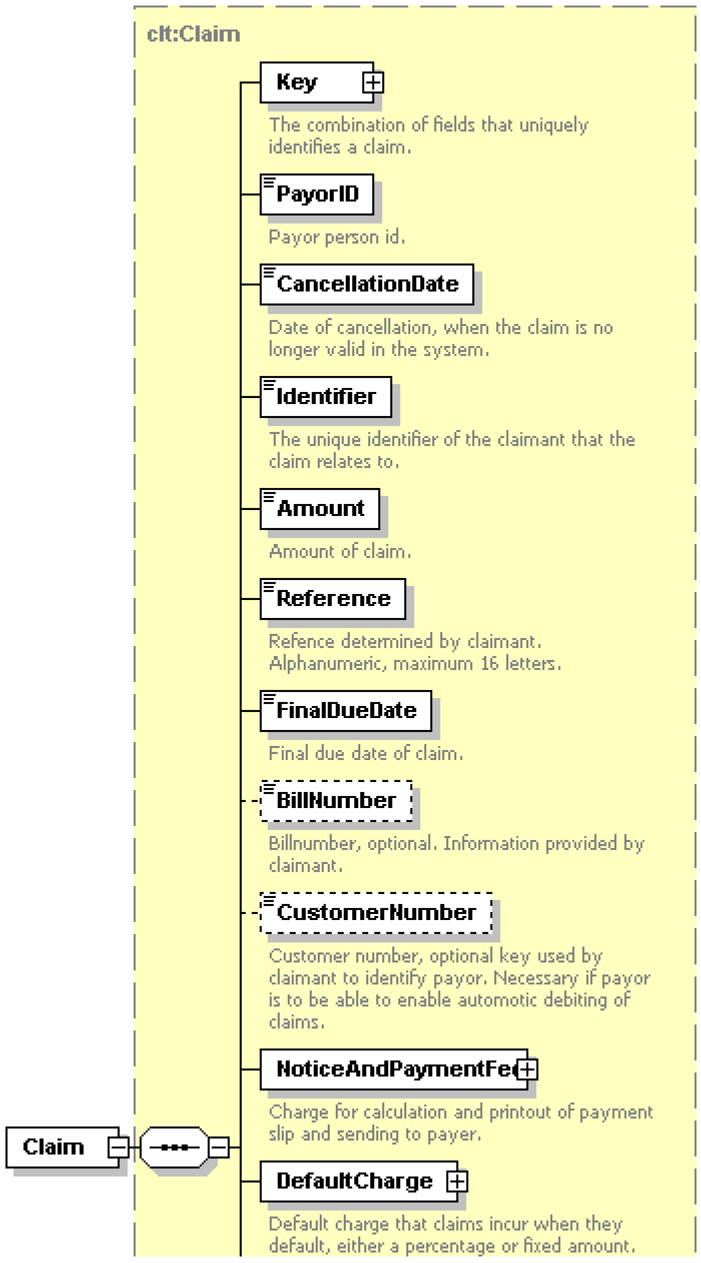
Sends in a list of keys for the claims to be cancelled. The claim key is the same as in the creation/modification of claims.

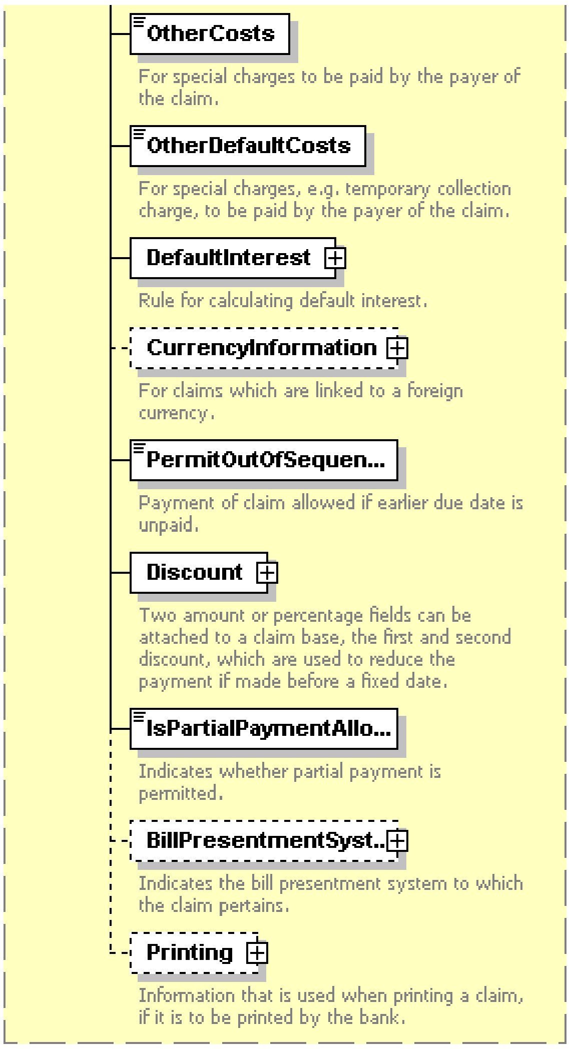

clt:ClaimKey
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
ClaimKey
A unique key of claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

The response to CancelClaims is the same as to creation/modification, i.e. CancelClaimsResponse that contains the string OperationID.

CreateClaim/AlterClaim

The creation and the modification of a single claim is the same as in Claims, except that here it is always a single claim that is being processed, not a list. The claim itself lookst the same, but the answer to creation/modification is a ClaimOperationResult.


clt:Claim
Claim
Key
The combination of fields that uniquely identifies a claim.
PayorID
Payor person id.
CancellationDate
Date of cancellation, when the claim is no longer valid in the system.
Identifier
The unique identifier of the claimant that the claim relates to.
Amount
Amount of claim.
Reference
Refence determined by claimant. Alphanumeric, maximum 16 letters.
FinalDueDate
Final due date of claim.
BillNumber
Billnumber, optional. Information provided by claimant.
CustomerNumber
Customer number, optional key used by claimant to identify payor. Necessary if payor is to be able to enable automotic debiting of claims.
NoticeAndPaymentFe
Charge for calculation and printout of payment slip and sending to payer.
DefaultCharge
Default charge that claims incur when they default, either a percentage or fixed amount.



For special charges to be paid by the payer of the claim.

For special charges, e.g. temporary collection charge, to be paid by the payer of the claim.

Rule for calculating default interest.

For claims which are linked to a foreign currency.

Payment of claim allowed if earlier due date is unpaid.

Two amount or percentage fields can be attached to a claim base, the first and second discount, which are used to reduce the payment if made before a fixed date.

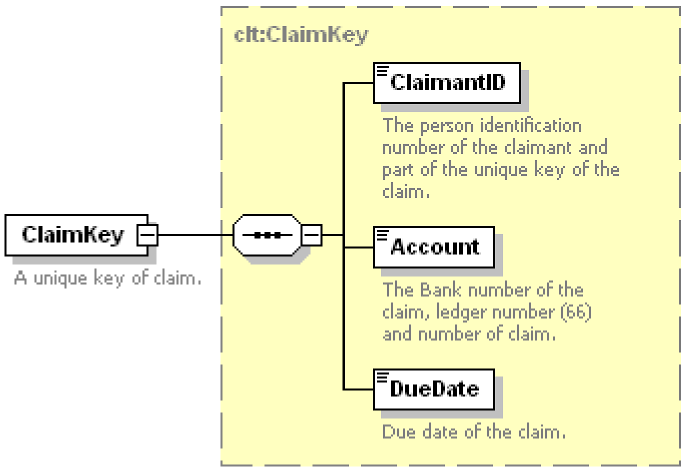
Indicates whether partial payment is permitted.

Indicates the bill presentment system to which the claim pertains.

Information that is used when printing a claim, if it is to be printed by the bank.

CancelClaim

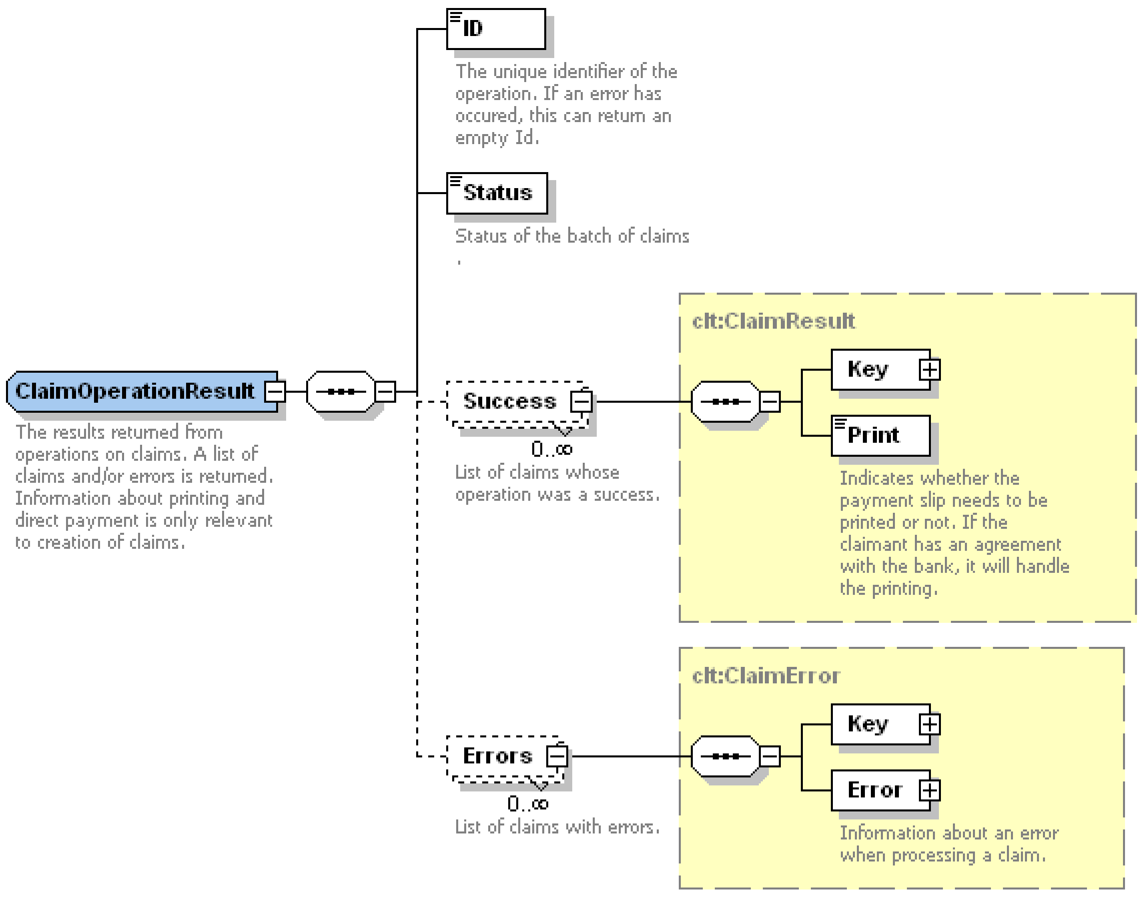
The cancellation of a claim is the same as in Claims, i.e. the key for the claim to be cancelled is sent, but here it is always a single claim that is being processed.


clt:ClaimKey
ClaimKey
A unique key of claim.
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

The answer to the cancellation is the same as in the create/alter operation, i.e. ClaimOperationResult.

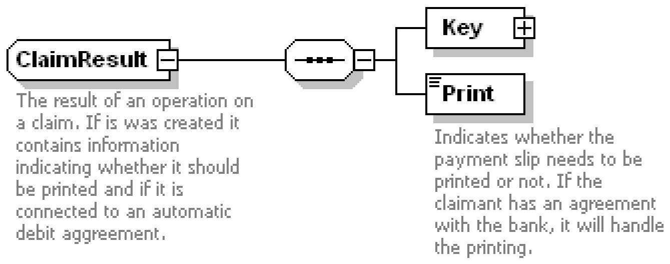
GetClaimOperationResult

A OperationID (string) is sent for the operation that information is to be collected about. The answer: **GetClaimOperationResultResponse** which contains **ClaimOperationResult**.



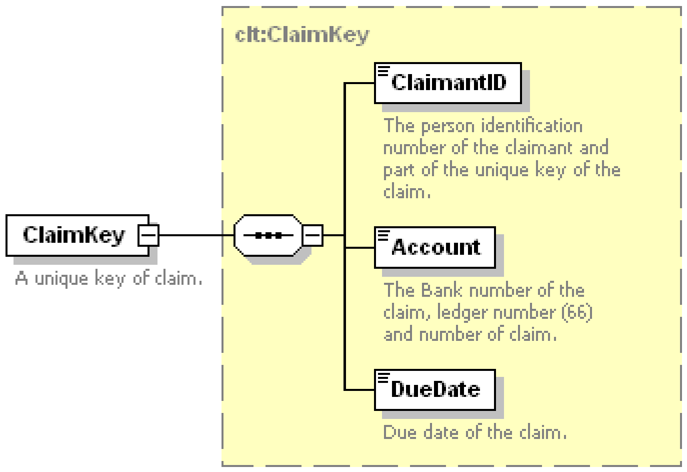
ClaimOperationResult
The results returned from operations on claims. A list of claims and/or errors is returned. Information about printing and direct payment is only relevant to creation of claims.
ID
The unique identifier of the operation. If an error has occured, this can return an empty Id.
Status
Status of the batch of claims
Success
0..∞
List of claims whose operation was a success.
clt:ClaimResult
Key
Print
Indicates whether the payment slip needs to be printed or not. If the claimant has an agreement with the bank, it will handle the printing.
Errors
0..∞
List of claims with errors.
clt:ClaimError
Key
Error
Information about an error when processing a claim.

Claims is a list of the claims that were successfully created, i.e. the claim key and information on whether it is to be printed or not.

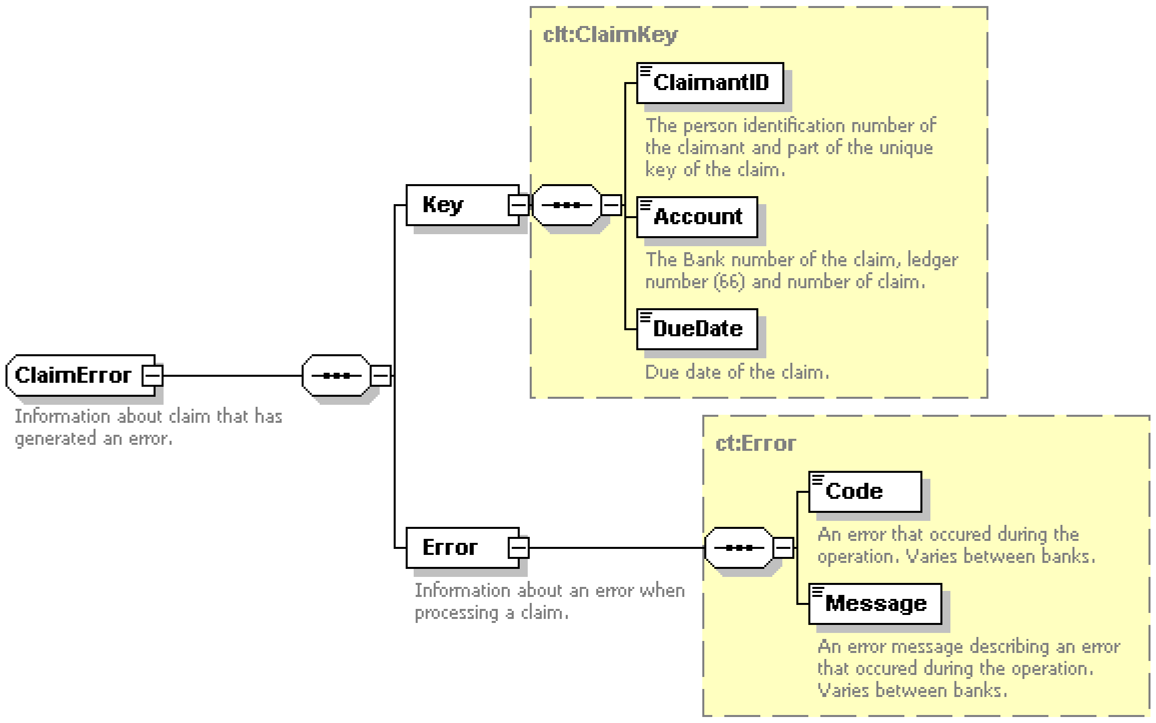


ClaimResult
The result of an operation on a claim. If is was created it contains information indicating whether it should be printed and if it is connected to an automatic debit aggreement.
Key
Print
Indicates whether the payment slip needs to be printed or not. If the claimant has an agreement with the bank, it will handle the printing.

The claim key is its unique identifier. All elements are obligatory.

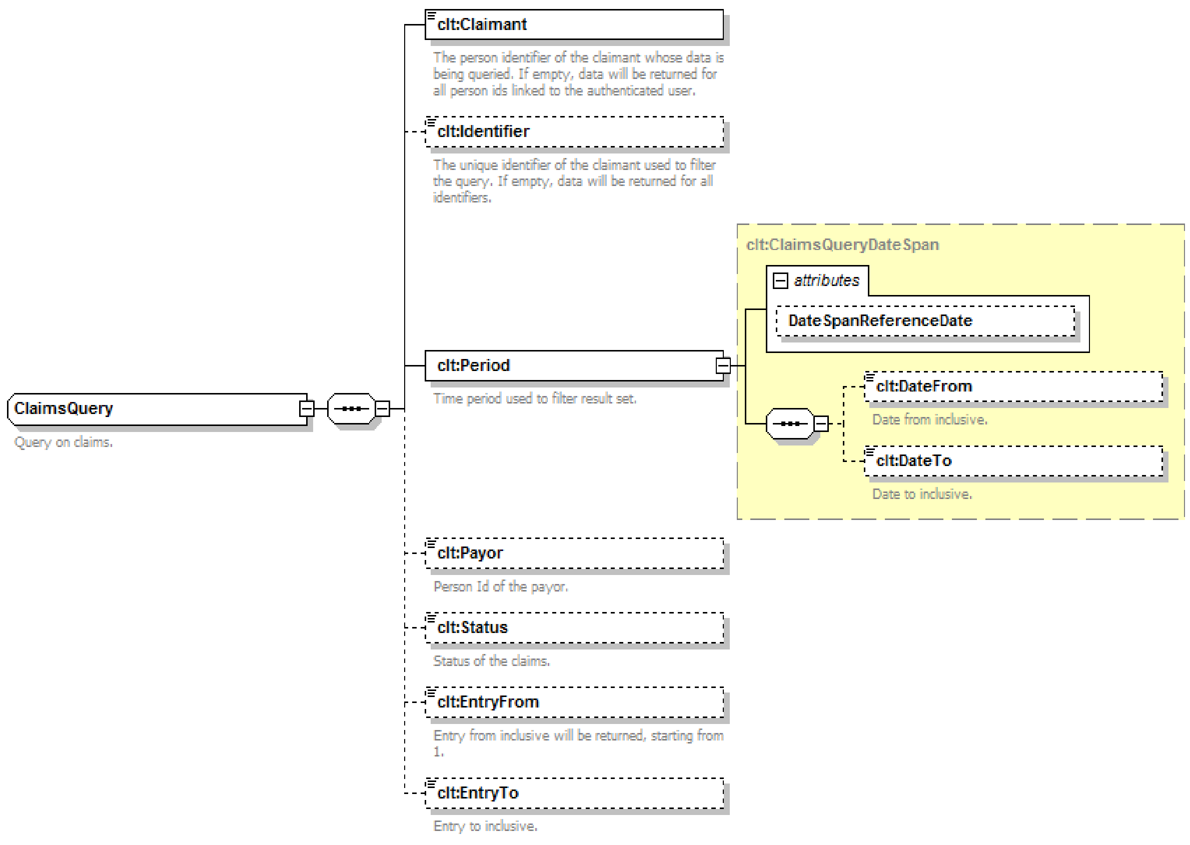

clt:ClaimKey
ClaimKey
A unique key of claim.
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

The list of errors in ClaimsResult is the same type as previously shown, e.g. in payments.

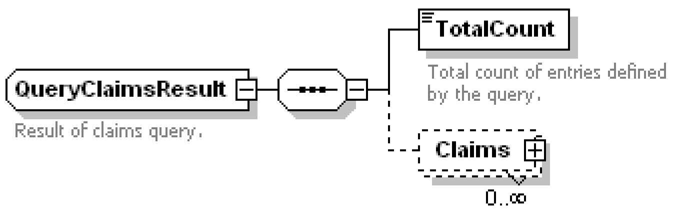

clt:ClaimKey
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Key
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.
ClaimError
Information about claim that has generated an error.
ct:Error
Code
An error that occured during the operation. Varies between banks.
Error
Information about an error when processing a claim.
Message
An error message describing an error that occured during the operation. Varies between banks.

QueryClaims

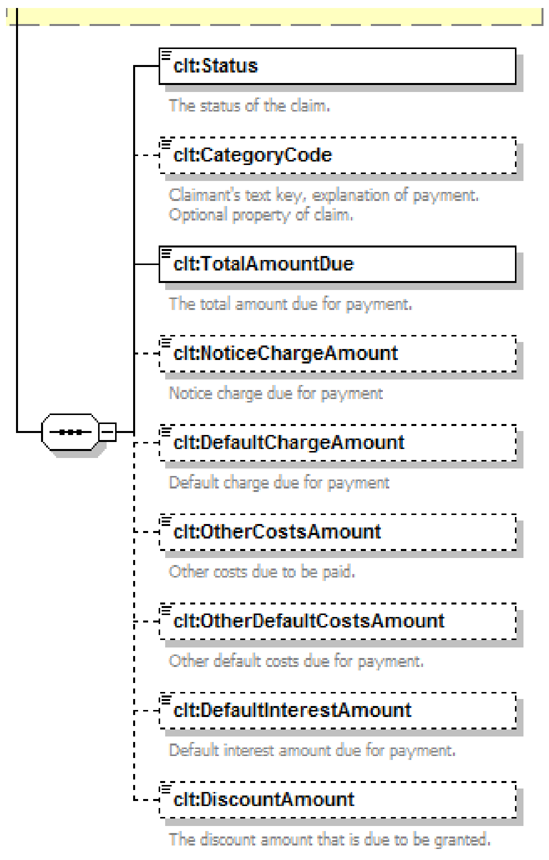
A query on claim status, which uses the ClaimsQuery type. The only obligatory element is Claimant, which is the claim owner. Other elements are mostly self-explanatory, except that it should be noted that when the result set is large only a certain number is returned and paging through the rest is called for, e.g. by specifying entries 501 through 1000 in the next query.


clt:Claimant
The person identifier of the claimant whose data is being queried. If empty, data will be returned for all person ids linked to the authenticated user.
clt:Identifier
The unique identifier of the claimant used to filter the query. If empty, data will be returned for all identifiers.
clt:ClaimsQueryDateSpan
attributes
DateSpanReferenceDate
clt:Period
Time period used to filter result set.
ClaimsQuery
Query on claims.
clt:DateFrom
Date from inclusive.
clt:DateTo
Date to inclusive.
clt:Payor
Person Id of the payor.
clt:Status
Status of the claims.
clt:EntryFrom
Entry from inclusive will be returned, starting from 1.
clt:EntryTo
Entry to inclusive.

The answer to QueryClaims is QueryClaimsResponse, which contains QueryClaimsResult. The Claims element will only contain a subset of the claims when the result set is large and TotalCount indicates how many remain to be retrieved.

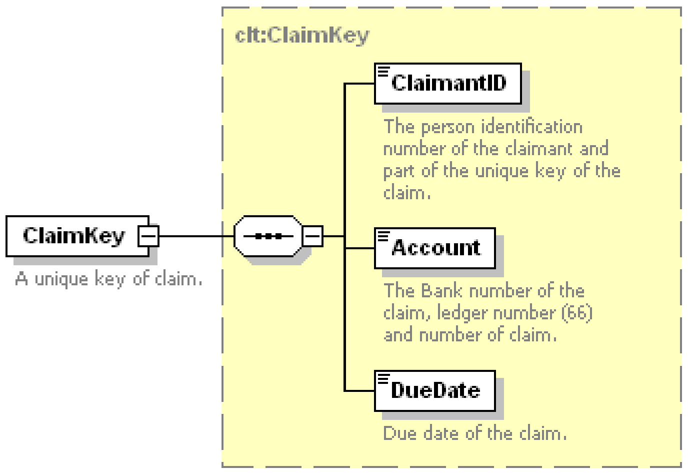


The claims in the list Claims are of the type ClaimInfo, but basically they are the same as the type Claim which has previously been described, with a few added elements. It includes more details about the costs that apply to the claim, e.g. default charge and discount. These additional elements are depicted below.

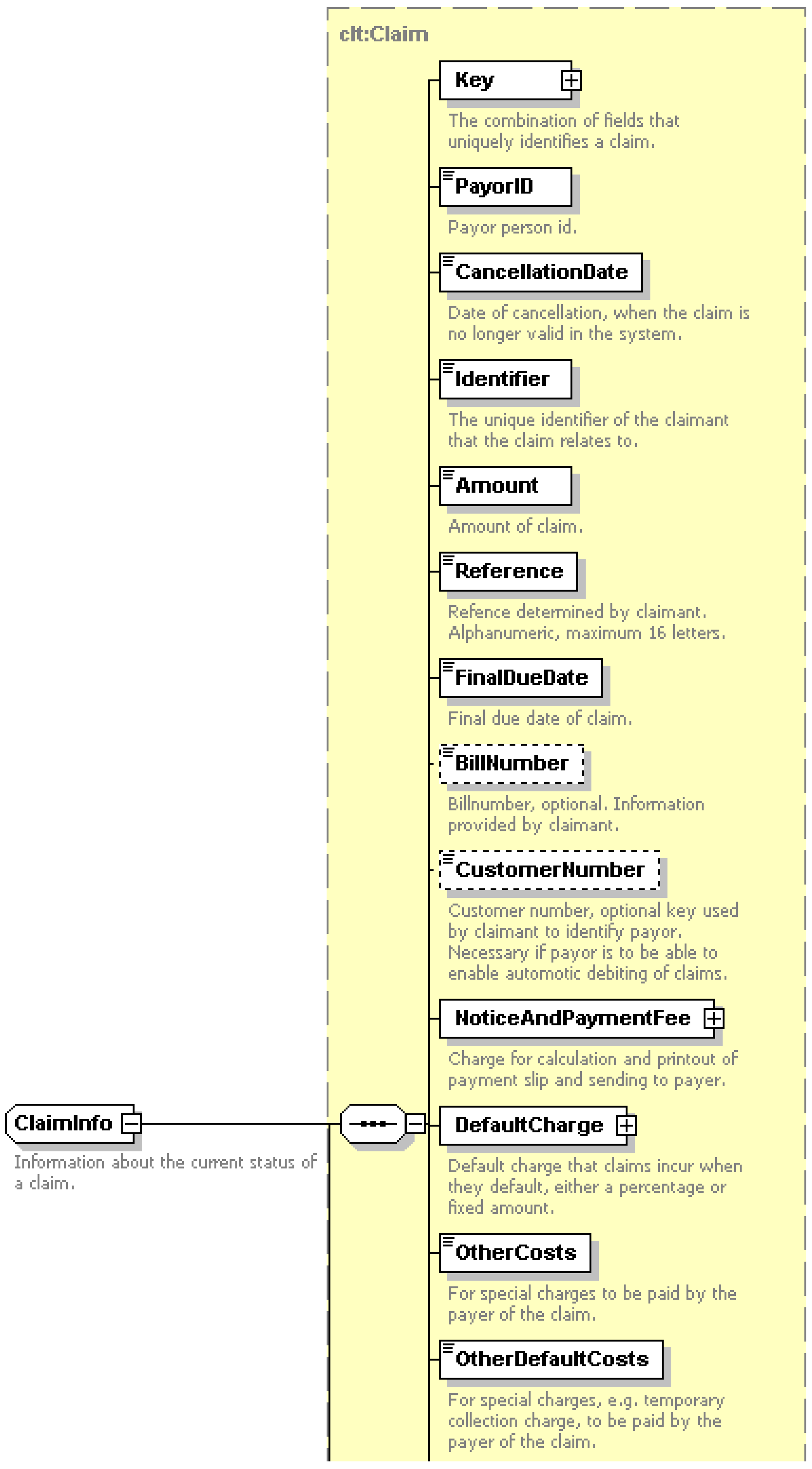

clt:Status
The status of the claim.
clt:CategoryCode
Claimant's text key, explanation of payment. Optional property of claim.
clt:TotalAmountDue
The total amount due for payment.
clt:NoticeChargeAmount
Notice charge due for payment
clt:DefaultChargeAmount
Default charge due for payment
clt:OtherCostsAmount
Other costs due to be paid.
clt:OtherDefaultCostsAmount
Other default costs due for payment.
clt:DefaultInterestAmount
Default interest amount due for payment.
clt:DiscountAmount
The discount amount that is due to be granted.

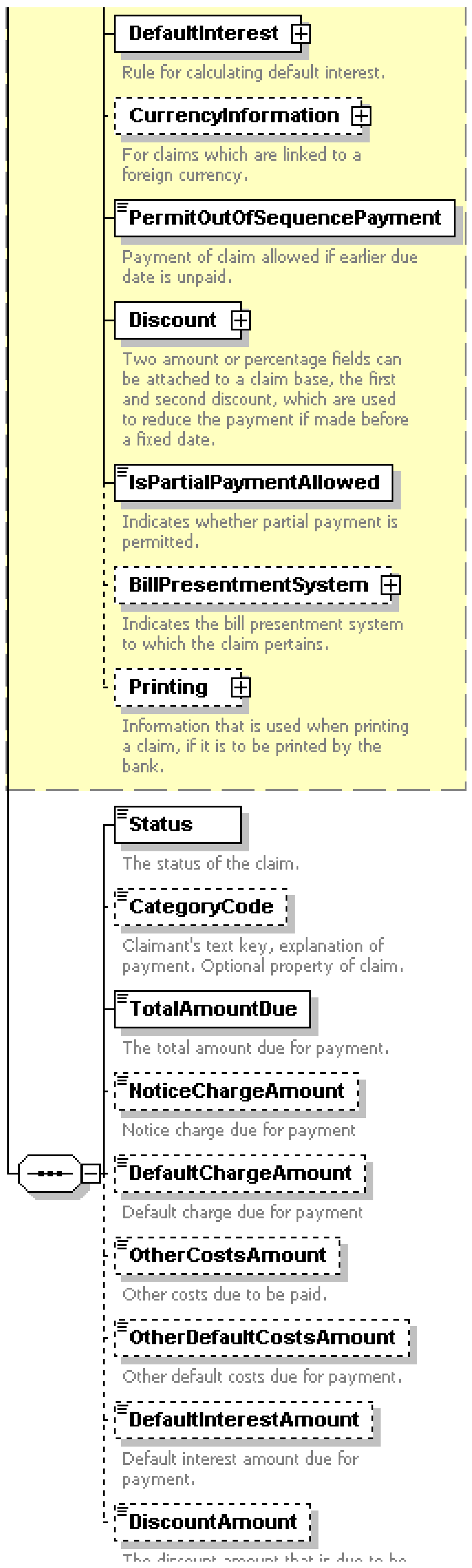
QueryClaim

A query on a single claim. Uses the claim key.


clt:ClaimKey
ClaimKey
A unique key of claim.
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

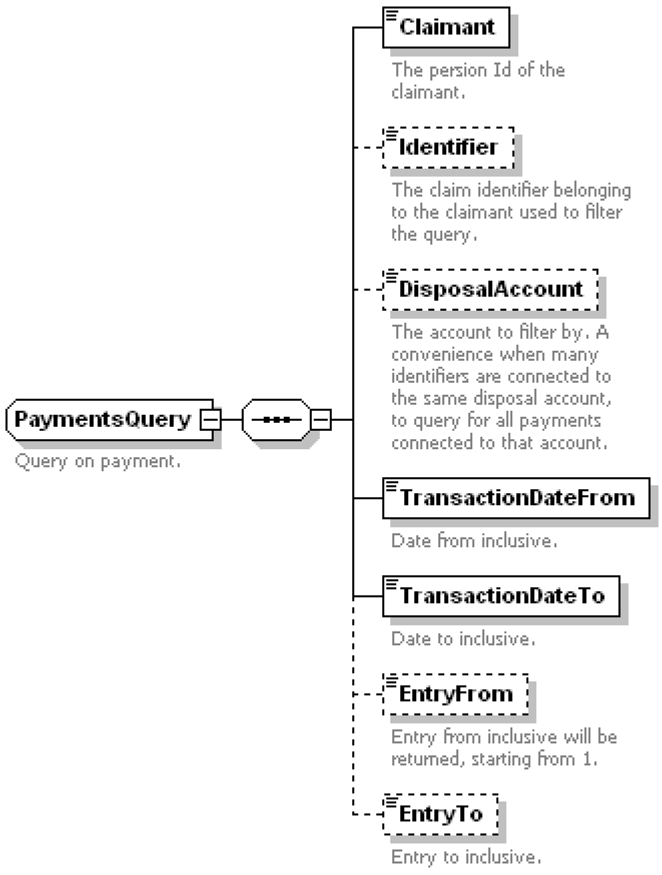
The answer to QueryClaim is QueryClaimResponse, which contains QueryClaimResult which is the type ClaimInfo, the same type as returned in QueryClaimsResult.


clt:Claim
ClaimInfo
Information about the current status of a claim.
Key
The combination of fields that uniquely identifies a claim.
PayorID
Payor person id.
CancellationDate
Date of cancellation, when the claim is no longer valid in the system.
Identifier
The unique identifier of the claimant that the claim relates to.
Amount
Amount of claim.
Reference
Refence determined by claimant. Alphanumeric, maximum 16 letters.
FinalDueDate
Final due date of claim.
BillNumber
Billnumber, optional. Information provided by claimant.
CustomerNumber
Customer number, optional key used by claimant to identify payor. Necessary if payor is to be able to enable automotic debiting of claims.
NoticeAndPaymentFee
Charge for calculation and printout of payment slip and sending to payer.
DefaultCharge
Default charge that claims incur when they default, either a percentage or fixed amount.
OtherCosts
For special charges to be paid by the payer of the claim.
OtherDefaultCosts
For special charges, e.g. temporary collection charge, to be paid by the payer of the claim.

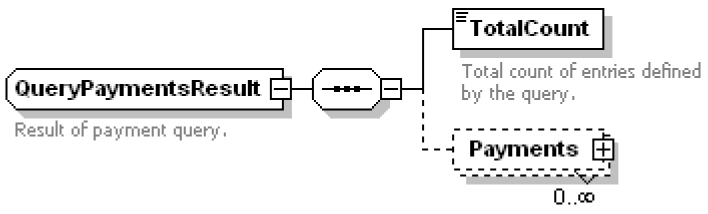

DefaultInterest
Rule for calculating default interest.
CurrencyInformation
For claims which are linked to a foreign currency.
PermitOutOfSequencePayment
Payment of claim allowed if earlier due date is unpaid.
Discount
Two amount or percentage fields can be attached to a claim base, the first and second discount, which are used to reduce the payment if made before a fixed date.
IsPartialPaymentAllowed
Indicates whether partial payment is permitted.
BillPresentmentSystem
Indicates the bill presentment system to which the claim pertains.
Printing
Information that is used when printing a claim, if it is to be printed by the bank.
Status
The status of the claim.
CategoryCode
Claimant's text key, explanation of payment. Optional property of claim.
TotalAmountDue
The total amount due for payment.
NoticeChargeAmount
Notice charge due for payment
DefaultChargeAmount
Default charge due for payment
OtherCostsAmount
Other costs due to be paid.
OtherDefaultCostsAmount
Other default costs due for payment.
DefaultInterestAmount
Default interest amount due for payment.
DiscountAmount

QueryPayments

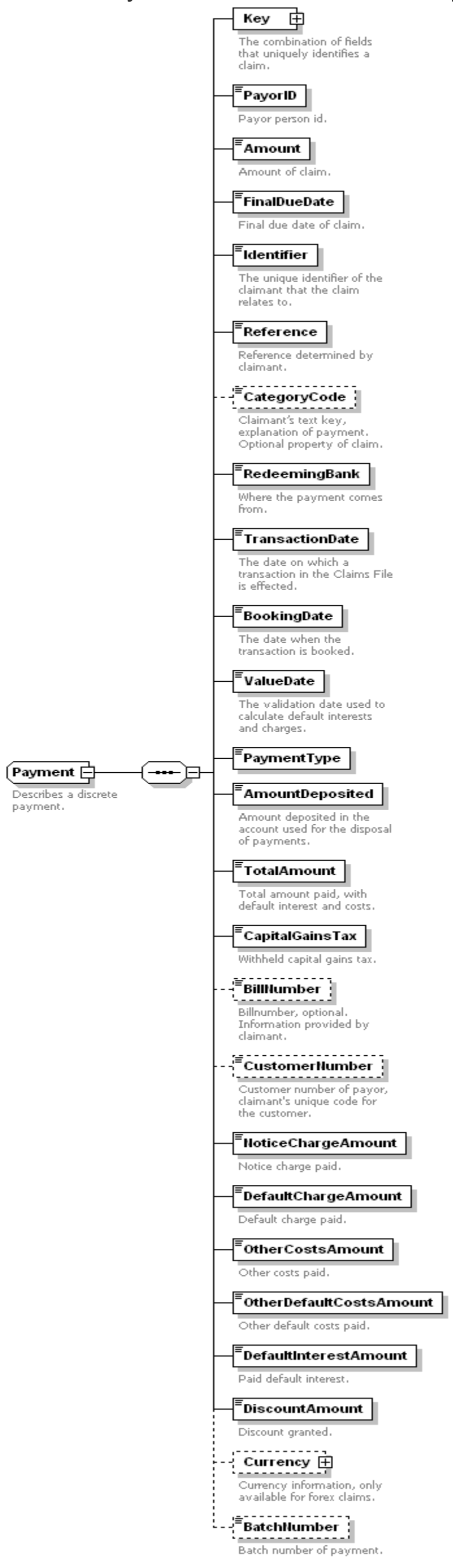
Uses the element query which is of type PaymentsQuery. It is possible to page through the result set, as previously done when querying Claims.


PaymentsQuery
Query on payment.
Claimant
The persion Id of the claimant.
Identifier
The claim identifier belonging to the claimant used to filter the query.
DisposalAccount
The account to filter by. A convenience when many identifiers are connected to the same disposal account, to query for all payments connected to that account.
TransactionDateFrom
Date from inclusive.
TransactionDateTo
Date to inclusive.
EntryFrom
Entry from inclusive will be returned, starting from 1.
EntryTo
Entry to inclusive.

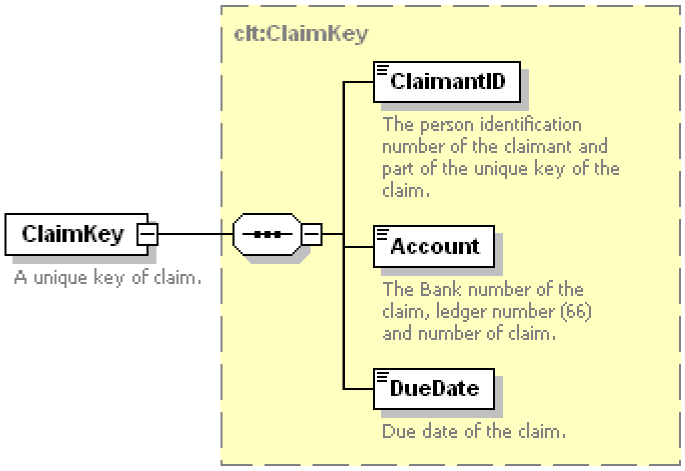
The mandatory time period set in the query is used for retrieving payments according to the transaction date (is. *hreyfingardagur*). As claim transactions are processed on the closing of each bank business day, the claimant will have to make sure to retrieve data for preceding weekends and bank holidays the next day it is accessible. Currently data for e.g. for weekends becomes available on Thursdays, the period from 21:00 on Friday to 21:00 the next Monday being processed together. The transaction date periods are normalized for the query to simplify retrieval, so transactions that occur after 21:00 are considered to belong to the next day. For example, payments occurring after 21:00 on the evening of Friday the 13th will be returned if TransactionDateFrom is set to the 14th. The answer to the QueryPayments query is QueryPaymentsResponse which has the element QueryPaymentsResult which is type QueryPaymentsResult. That contains a list of payments, as well as the total number of payments returned.


QueryPaymentsResult
Result of payment query.
TotalCount
Total count of entries defined by the query.
Payments
0..∞

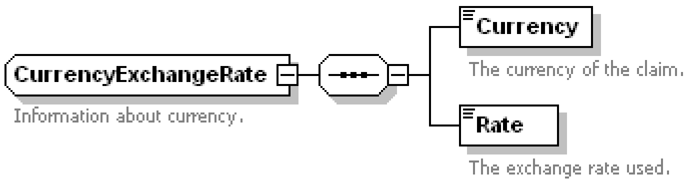
The list Payments contains a list of Payment.


Payment
Describes a discrete payment.
Key
The combination of fields that uniquely identifies a claim.
PayorID
Payor person id.
Amount
Amount of claim.
FinalDueDate
Final due date of claim.
Identifier
The unique identifier of the claimant that the claim relates to.
Reference
Reference determined by claimant.
CategoryCode
Claimant's text key, explanation of payment. Optional property of claim.
RedeemingBank
Where the payment comes from.
TransactionDate
The date on which a transaction in the Claims File is effected.
BookingDate
The date when the transaction is booked.
ValueDate
The validation date used to calculate default interests and charges.
PaymentType
AmountDeposited
Amount deposited in the account used for the disposal of payments.
TotalAmount
Total amount paid, with default interest and costs.
CapitalGainsTax
Withheld capital gains tax.
BillNumber
Billnumber, optional. Information provided by claimant.
CustomerNumber
Customer number of payor, claimant's unique code for the customer.
NoticeChargeAmount
Notice charge paid.
DefaultChargeAmount
Default charge paid.
OtherCostsAmount
Other costs paid.
OtherDefaultCostsAmount
Other default costs paid.
DefaultInterestAmount
Paid default interest.
DiscountAmount
Discount granted.
Currency
Currency information, only available for forex claims.
BatchNumber
Batch number of payment.

The key in Payment is the same as previously shown, i.e. the claim key.


clt:ClaimKey
ClaimKey
A unique key of claim.
ClaimantID
The person identification number of the claimant and part of the unique key of the claim.
Account
The Bank number of the claim, ledger number (66) and number of claim.
DueDate
Due date of the claim.

In addition, currency information is available for currency claims, but that type (CurrencyExchangeRate) only contains information about the currency and its rate.



IcelandicOnlineSecondaryCollectionClaims

Secondary collection agencies generally have access to the same operations on the claims being collected as original claimants, with the exception of claim creation. In addition, several specific operations are added for secondary collection companies as well as minor modifications of the queries.

AlterClaims / CancelClaims / AlterClaim / CancelClaim / GetOperationResult / QueryClaim

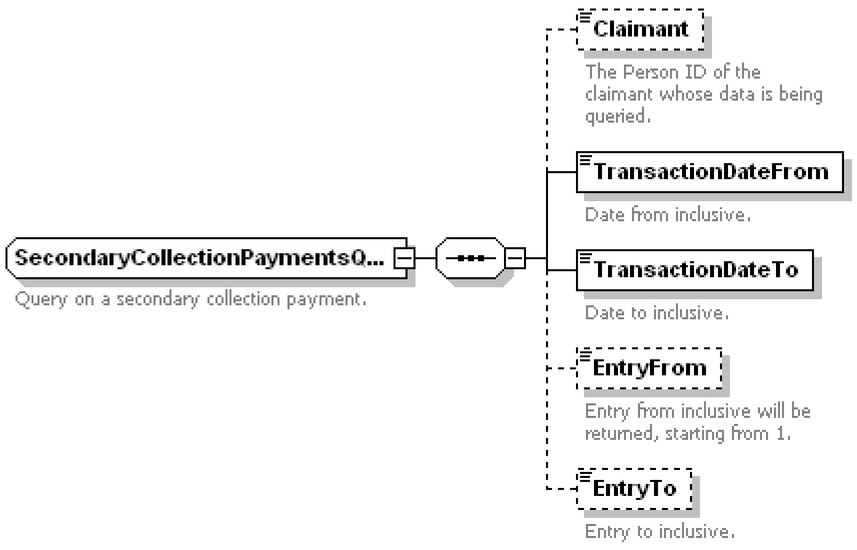
These operations differ from those in the main claim service only by the permissions the collection agency has to manipulate and query claims belonging to other claimants but having been assigned to the agency for collection.

QueryClaims

The query returns information about claims the secondary collection agency is responsible for collecting. It uses the same schema as the QueryClaims operation in the main claim service. Additionally collection agencies can either leave the Claimant field empty in order to query all collection claims or filter by a specific claimant.
The collection agency uses the main claim service to query its own claims.

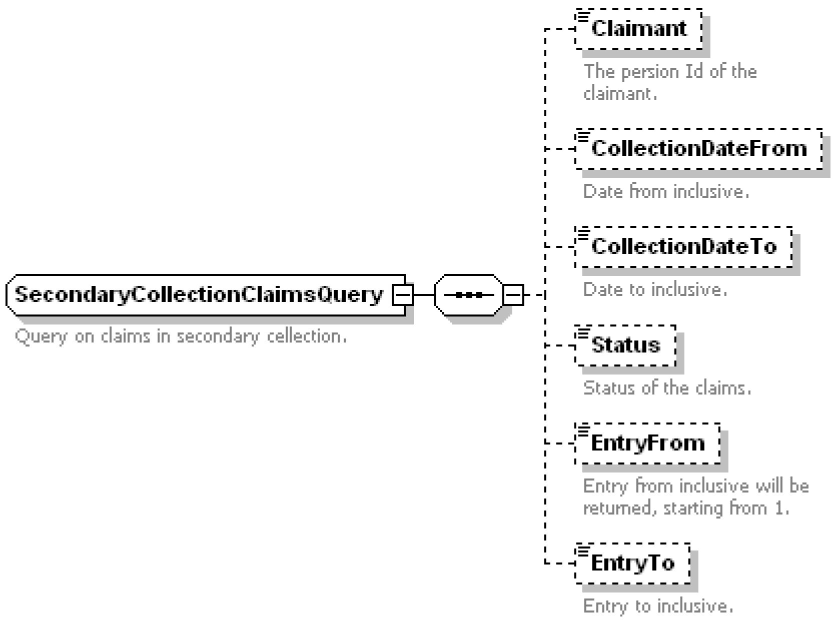
QuerySecondaryCollectionPayments

A method for secondary collection agencies to query about payments for claims they are collecting. Fetches the payments that have been processed, where it is possible to retrieve claims filtered by claimants. If no claimant is specified, all payments within the given time period are fetched.


SecondaryCollectionPaymentsQ...
Query on a secondary collection payment.
Claimant
The Person ID of the claimant whose data is being queried.
TransactionDateFrom
Date from inclusive.
TransactionDateTo
Date to inclusive.
EntryFrom
Entry from inclusive will be returned, starting from 1.
EntryTo
Entry to inclusive.

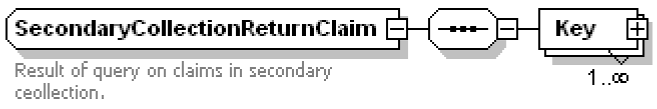
QuerySecondaryCollectionClaims

A query that returns claims that have been assigned to the secondary collection agency during a certain time period. Used mainly for retrieving information about new claims being assigned to the agency for collection.
All elements in the query are optional, if none are used, all claims that have come into collection for this company will be returned. As with the payment query, it is possible to narrow the search down to individual claimants.


SecondaryCollectionClaimsQuery
Query on claims in secondary collection.
Claimant
The persion Id of the claimant.
CollectionDateFrom
Date from inclusive.
CollectionDateTo
Date to inclusive.
Status
Status of the claims.
EntryFrom
Entry from inclusive will be returned, starting from 1.
EntryTo
Entry to inclusive.

SecondaryCollectionReturnClaim

Secondary collection companies can return claims that have reached the secondary collection status. It uses a list of claim keys for the claims that are to be returned.


SecondaryCollectionReturnClaim
Result of query on claims in secondary ceollection.
Key
1..∞