Operation Manual

Way4 Account Schemes

03.52.30

27.01.2021

Contents

This document is intended for bank or processing center employees responsible for configuring Way4 and describes Product creation and configuration.

While working with this document, it is recommended that users refer to the following resources from the Way4 documentation series:

- Way4 Global Parameters
- Way4 Dictionaries
- Currency Conversion
- Standing Payment Orders
- Daily Procedures
- Way4 Service Packages
- Interest Accrual
- Events
- Balance Types
- WAY4™ Advanced Tariff Management
- Loan Loss Reserves
- Way4™ Accounting

The following notation is used in the document:

- Field labels in screen forms are shown in *italics*.
- Screen form button labels are shown in square brackets; for example [Approve].
- Sequences for selecting user menu items are shown using arrows as follows: "Configuration Setup → Contract Types".

Warnings about potentially hazardous situations or actions are marked with a special icon and highlighted.

Information about important features, additional options, or the best use of certain system functions is marked with a special icon and highlighted.

1 Terms and Definitions

WAY4 is used to resolve the following tasks:

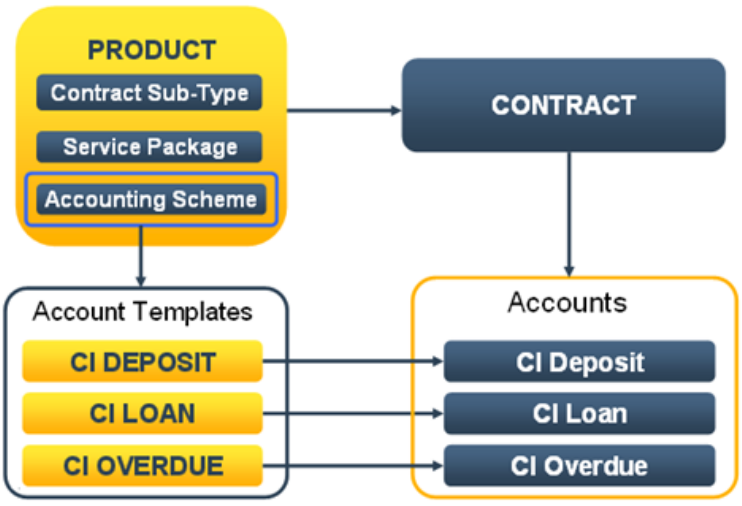
- Issue and acquire payment cards of individuals and corporate clients.
- Acquire merchants.
- Account and process financial transactions, including card transactions.
- Service current, deposit and loan accounts of individuals.

The key system object that allows the above functions to be performed is the contract. A contract is an accounting object that regulates the relationship between a bank and a settlement party: a bank client (including merchants) or a bank branch. Three categories of contracts are used in the system: issuing/acquiring contracts, card contracts, and device contracts.

Financial transactions are registered in the system between contracts. Contracts regulate transaction rules (allowed and forbidden transactions, transaction fees), a set of contract accounts, account interest rates, etc.

Contract properties are determined by three main parameters:

- Contract type/subtype – determines the "nature" of Mastercard or Visa, magnetic stripe or smart card), device contract (ATM, POS terminal, imprinter), or issuing/acquiring contract (a set of contract accounts and rules for working with them for an individual, a legal entity, or a bank branch).
- Service Package – contains a list of transactions, fee parameters, transaction processing rules, and usage limiters.
- Account Scheme – determines a contract's accounts, their properties and relations between them (see the figure).



Relations between system objects

WAY4 allows for registering Products – sets of main contract parameter values (contract subtype, Service Package, and Account Scheme). Products are used to optimize the setup of contract properties.

2 Account Schemes

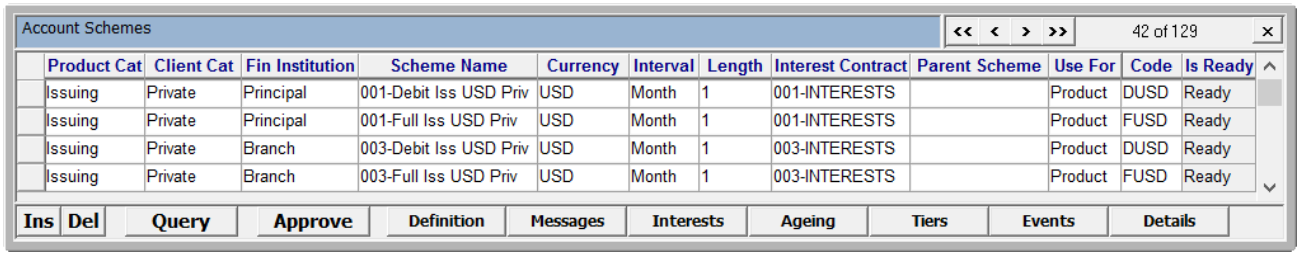
An Account Scheme consists of account templates used to:

- Create contract accounts.
- Set up relations between accounts – cash flow rules used when interest is accrued, a due date arrives, or account balances change.
- Set up interaction of accounts with the General Ledger.
  For more information, see the document "WAY4™ Accounting".
- Account template settings in an Account Scheme makes it possible to implement the following:
  - Accrual of interest on account balances (see the section "Interest Properties").
  - Amount normalization (see the section "Limit Normalization") – process where funds are automatically redistributed among the contract's accounts when their balances change.
  - Due normalization (see the section "Ageing") – process where funds are automatically transferred between accounts when a specific time limit is reached, for example, when a loan is transferred to an overdue account with a higher interest rate.
  - Account numeration (see the section "GL Properties").

2.1 "Account Schemes" Form

To set up a new Account Scheme, select "Full → Configuration Setup → Products → Account Schemes" from the user menu.

This will open the "Account Schemes" grid form (see figure).



Form for entering and viewing Account Scheme data

The "Account Scheme" form is used to access the complete list of registered Account Schemes that belong to all financial institutions and contract categories. For instance, it is convenient to use this form to copy a scheme to another financial institution. To set up a single Account Scheme, it is more convenient to use special forms with preliminary data filtering (see the section "Special Forms for Working with Account Schemes").

The "Account Scheme" form contains fields for specifying main parameters of an Account Scheme. Form fields are described in the section "Main Account Scheme Parameters".

The form also contains the following control buttons:

- The [Approve] button is used to approve an Account Scheme. For more information, see the section "Account Scheme Approval".
- The [Definitions] button is used to access Account Scheme templates (see the section "Full Information about Account Scheme Templates").
- The [Interests] button is used to access the "Interests for <name of Account Scheme>" form, where interest accrual parameters for all templates of an Account Scheme can be specified (the parameters can also be specified in the "Interest Properties" group of fields in the form containing full account template information – see the section "Full Information about Account Scheme Templates").
- The [Ageing] button is used to access the "Ageing for <name of Account Scheme>" form, where due normalization parameters for all templates of an Account Scheme can be specified (the parameters can also be specified in the "Ageing" group of fields in the form containing full account template information – see the section "Full Information about Account Scheme Templates").
- The [Tiers] button is used to access the "Tiers for <name of Account Scheme>" form, where limit normalization parameters for all templates of an Account Scheme can be specified (the parameters can also be specified in the "Limit Normalization" group of fields in the form containing full account template information – see the section "Full Information about Account Scheme Templates").
- The [Events] button is used to set up Events for changing a contract's behavior type, for changing a contract's Account Scheme, for changing the interest rate on a contract's account (see the section "Configuring Events").
- The [Details] button is used to access additional parameters of an Account Scheme (see the section "Additional Account Scheme Parameters").

2.1.1 Main Account Scheme Parameters

2.1.1.1 Product Cat

The *Product Cat* field determines the Product type for which this Account Scheme may be used.

2.1.1.2 Client Cat

The *Client Cat* field determines a client type, selected from a drop-down list, where "Private" is an individual, "Commercial" is a legal entity, and "Accountant" is a bank branch.

2.1.1.3 Fin Institution

The *Institution* field contains the financial institution for which the Account Scheme is generated. Account Schemes generated for a financial institution will be available when contracts are registered in this financial institution.

2.1.1.4 Scheme Name

The name of an Account Scheme is entered in the *Scheme Name* field. It is recommended that the following name format be used: NNN-name, where NNN is the last three digits of the financial institution's number registered in the database (see the *Branch Code* field in the "Full → Configuration Setup → Main Tables → Financial Institutions" form).

2.1.1.5 Parent Scheme

The *Parent Scheme* field determines the Account Scheme whose properties are inherited by the current Scheme. This is a drop-down list of all Account Schemes of all financial institutions registered in the system.

2.1.1.6 Used For

There are two types of Account Scheme:

- Main Account Schemes, used when registering contracts in WAY4 – for these Account Schemes, the value of the *Used For* field is either empty or "Product".
- Included Account Schemes – for these Account Schemes, the "Included" value is specified in the *Used For* field. Included Account Schemes are attached to main Account Schemes. Included Account Schemes are used when a set of accounts must be added to several Account Schemes. For more information, see the section "Working with Included Account Schemes".

2.1.1.7 Currency

It is recommended that users select a contract currency in the *Currency* field. For instance, this currency will be used to calculate the amount available for issuing contracts. The currency will also be used to generate contract reports.

2.1.1.8 Interval

The *Interval* field determines the time unit in which the length of a billing cycle will be measured.

2.1.1.9 Length

The *Length* field determines the length of a billing cycle in units specified in the *Interval* field.

2.1.1.10 Code

The *Code* field contains a special code used when external files are imported into WAY4. The code allows imported data to be matched with the correct Account Scheme.

2.1.1.11 Interest Contract

The *Interest Contract* field determines the bank contract containing interest accrual accounts.

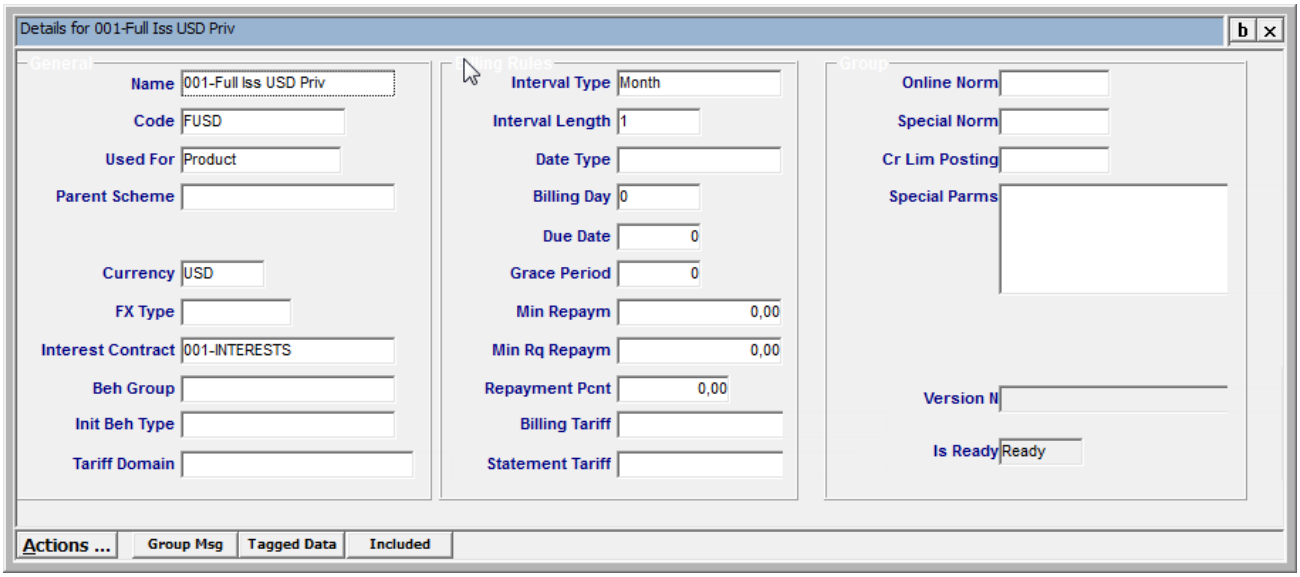
2.1.1.12 Is Ready

The *Is Ready* field shows whether changes made to this Account Scheme have been approved.

- The field contains the "Ready" value if changes made to the Account Scheme have been approved.
- The field contains the "Not Ready" value if changes made to the Account Scheme have not been approved.
- For information about Account Scheme approval, see the section "Account Scheme Approval".

2.1.2 Additional Account Scheme Parameters

To access additional information about an Account Scheme, use the form opened by clicking the [Details] button in Account Scheme grid forms.



Form "Details for <name of Account Scheme>

The form with additional Account Scheme data contains the following control buttons:

- Clicking the [Actions] button opens the following context menu items:
- [Check] – verify Account Scheme parameters (see the section "Checking Account Schemes").
- [Duplicate] – copy the Account Scheme (see the section "Copying Account Schemes").
- [Save Tagged Data] – save the tags entered in the *Special Parms* field (see the section "Entering and Editing Tags").
- The [Group Msg] button is used to set up client message templates (see the section "Configuring Message Templates (Group Msg)").
- The [Tagged Data] button is used to optimise work with the *Special Parms* field. Clicking this button opens a grid form for entering and editing tags in the *Special Parms* field (see the section "Entering and Editing Tags").
- The [Included] button is used to work with included Account Schemes. For more information, see the sections "Used For" and "Working with Included Account Schemes".

This form contains several fields absent from the parent Account Scheme forms.

For information on the use of fields *Grace Period*, *Min Repaym*, and *Min Rq Repayment* on the Account Scheme level, please contact Customer Support.

2.1.2.1 FX Type

Conversion rate type. For more details on currency conversion in Way4, see the document "Currency Conversion".

2.1.2.2 Interest Scheme

This parameter is used to calculate the daily interest rate using an annual interest rate. The *Interest Scheme* parameter in an Account Scheme redefines the value of the parameter with the same name in a financial institution. The *Interest Scheme* parameter in an Account Scheme can be redefined in an account template.

The field can have one of the following values:

- "Default – when calculating interest, the value of the financial institution's *Interest Scheme* parameter will be used.
- "Actual 365/366" – to determine the daily interest rate, the length of a year will be equal to the actual number of calendar days in the year (either 365 or 366 days).
- "360" – to calculate the daily interest rate, the length of a year will be affected by the value of the USE_MONTH_WEIGHT global parameter (see the document "Way4 Global Parameters" and sections "Number of Days in a Year" and "Determining a Daily Interest Rate" of the document "Interest Accrual").
- "-360" – to calculate the daily interest rate, the number of calendar days in a month will be considered to be 30, and the number of calendar days in a year will be considered to be 360. For more information, see the sections "Number of Days in a Year" and "Determining a Daily Interest Rate" of the document "Interest Accrual".
- "Fixed 365" – fixed value for the number of calendar days in the year (365).
- "Fixed 366" – fixed value for the number of calendar days in the year (366).
- "360 with USE_MONTH_WEIGHT=Y" – to determine the daily interest rate, each month is considered to have the same weight, equal to 1/12 of a year. For example, more interest is accrued for one day of February than is accrued for the same amount on one day in January.
- "360 with USE_MONTH_WEIGHT=N" – for calculating the daily interest rate it is assumed there are 360 days in a year. Months are considered to have different weights depending on the number of days in the month. For example, the same amount of interest is accrued for one day in February as is accrued for the same amount on one day in January.
  For more information, see the sections "Number of Days in a Year" and "Determining a Daily Interest Rate" of the document "Interest Accrual".

2.1.2.3 Behavior Group

The group of possible Behavior Types for contracts using this Account Scheme.

2.1.2.4 Init Beh Type

The Behavior Type assigned at the time of registering a contract using this Account Scheme.

2.1.2.5 Tariff Domain

For each Account Scheme, one of the tariff domains registered in WAY4 can be specified the *Tariff Domain* field. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module.

> The tariff management module is not included in the basic configuration of WAY4 and is provided according to an additional agreement with OpenWay.

2.1.2.6 Date Type

The *Date Type* field determines the time shift of the end of a billing cycle for the number of days specified in the *Billing Day* field. Possible values:

- Empty (null) – the shift is not applied.
- "From Last Day" – the shift is calculated from the last day of the billing cycle.
  For example, the *Interval* field is set to "Month", the last day of the current billing cycle should be 31 January, and the *Billing Day* field is set to "-1". In this case, the actual last day of the billing cycle will be 30 January. When the *Billing* Day field is set to "1", the actual last day of the billing cycle will be 01 February.
- "From Last Weekend" – the shift is calculated from the last weekend day of the billing cycle. The system differentiates between business days and weekends using a business calendar (see the section "Business Calendar" of the document "WAY4™ Dictionaries").
  For example, the *Interval* field is set to "Month", and the current month is January. 26 January is Friday, 27 and 28 January are weekends, and 29 January is Monday. If the *Billing Day* field is set to "-1", the actual last day of the billing cycle will be 27 January. If the *Billing Day* field is set to "1", the actual last day of the billing cycle will be 29 January. If the *Billing Day* field is set to "4", the actual last day of the billing cycle will be 01 February.
- "Fixed Day of Week" – the *Billing Day* field specifies the day of the week when the next billing cycle will start.
  Note that in WAY4 days are assigned fixed numbers according to European numeration conventions: Monday is "1", and Sunday is "7".
  For example, the *Interval* field is set to "Month", the next month is February, and 01 February is Thursday. If the *Billing Day* field is set to "5", the actual last day of the billing cycle will be Friday, 02 February. If the *Billing Day* field is set to "1", the actual last day of the current billing cycle will be Monday, 05 February.
- "From Last Wrk Day" – the shift is calculated from the last business day of a month. The system differentiates between business days and weekends using a business calendar (see the "Business Calendar" section of the document "WAY4™ Dictionaries).

For example, the *Interval* field is set to "Month", and the current month is March. 30 March is Friday, 31 March is Saturday.
If the *Billing Day* field is set to "0", the actual last day of the billing cycle will be Friday, 30 March.
If the *BillingDay* field is set to "-1", the actual last day of the billing cycle will be Thursday, 29 March.
If the *Billing Day* field is set to "1", the actual last day of the billing cycle will be 2 April. If the *Billing Day* field is set to "4", the actual last day of the billing cycle will be Thursday, 5 April.

- "From Open Date" – when the last day of the billing cycle is calculated, the contract's opening date is considered.
  For example, the *Interval* field is set to "Month", and the contract opening date is 14 March. The last day of the billing cycle will be 13 April.
- "Fixed Day of Month" – the *Billing Day* field shows a specific day of the month when a billing cycle will end.
  For example, the *Interval* field is set to "Month", and the *Billing Day* field is set to "15". Then the last day of a billing cycle will be the 15th day of every month.
- "Custom" – the time shift is calculated using a custom procedure.

2.1.2.7 Billing Day

The *Billing Day* field makes it possible to specify the length of a time shift of the end of a billing cycle.

- *Billing Day* > 0 – the end of a billing cycle is shifted forward for the number of days specified in the *Billing Day* field.
- *Billing Day* < 0 – the end of a billing cycle is shifted backward for the number of days specified in the *Billing Day* field.

If the *Date Type* field is set to "Fixed Day of Week", the *Billing Day* field contains a specific day of the week.

If the *Date Type* field is set to "Fixed Day of Month", the *Billing Day* field contains a specific day of the month.

This field remains for backward compatibility. It is recommended to use date schemes for calculating billing cycle end dates (see the document "Contract Functional Dates").

Any changes in rules for calculating billing dates must be approved by OpenWay and must have been tested in advance.

2.1.2.8 Due Date

This field contains the number of days used to calculate the date shown in account statements as the scheduled due date of a loan or loan interest.

Example.

For example, an Account Scheme has the following parameters:

- A billing cycle is one month.
- The *Due Date* field is set to "16".
- The Account Scheme contains, among others, the following account templates:
- "Cl Loan" account template with the following parameters:
- *Due Type* = "End Cycle Due"
- *Due Template* = "Cl Paym Due"
- *Due Period* = "0".
- "Cl Paym Due" with the following parameters:
- "Payment Due" category (see the *Category* field in the "Full → Configuration Setup → Accounting Setup → Account Types" form).
- *Due Type* = "Payment Due"
- *Due Template* = "Cl OVD"
- *Due Period* = "20".

On the last day of the month when the debt arose, funds from the "Cl Loan" account are transferred to the "Cl Paym Due" account according to due normalization rules. The date specified in the statement as the payment due date is determined as follows: the value specified in the *Due Date*parameter is added to the opening day of the next billing cycle. If this date falls on a weekend, the statement will show the last business before the calculated date.

Therefore, in our example, the statement will contain a message that the client must repay the debt by the 17th day of the next month or, if the 17th is a weekend, the last business day before the 17th.

> Note that the value of the *Due Date* field does not affect due normalization, that is, in our example funds are transferred from the "Cl Paym Due" account to the "Cl OVD" account at the end of the 20th day.

2.1.2.9 Grace Period

This field is used to calculate minimum payment amounts for credit products. For more details, please contact Customer Support.

2.1.2.10 Min Repaym

This field is used to calculate minimum payment amounts for credit products. For more details, please contact Customer Support.

2.1.2.11 Min Rq Repaym

This field is used to calculate minimum payment amounts for credit products. For more details, please contact Customer Support.

2.1.2.12 Billing Tariff

Field with a drop-down list of registered tariff types with the "Billing Scheme" role. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module. Used in calculating the end date of a billing cycle, minimum payment for credit products.

> It is recommended to use date schemes for calculating billing cycle end dates (see the document "Contract Functional Dates").

> The tariff management module is not included in the basic configuration of WAY4 and is provided according to an additional agreement with OpenWay.

2.1.2.13 Statement Tariff

Field with a drop-down list of registered tariff types with the "Ageing" role. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module. Used when calculating the minimum payment for credit products.

> The Advanced Tariff Management module is not included in the WAY4 basic configuration and is delivered according to an additional agreement with OpenWay.

2.1.2.14 Online Norm

The field is used to specify the order of normalization of accounts created using this Account Scheme.

The field contains a drop-down list of possible values, and the following rules are applied:

- When the ONLINE_NORMALIZATION global parameter is set to one of the documented values (see the document "WAY4™ Global Parameters"), normalization is performed according to the specified value of the global parameter regardless of the value of the *Online Norm* local parameter (including cases where the *Online Norm* field is left blank).
- The *Online Norm* local parameter only redefines the ONLINE_NORMALIZATION global parameter if the global parameter is set to an undocumented value, and:
- If the *Online Norm* field contains the value "Yes", amount normalization between accounts of this Account Scheme is performed immediately during macrotransaction posting.
- If the *Online Norm* field contains the value "No", account normalization is not performed during macrotransaction posting; in this case, a special DB Manager user menu item performing contract account normalization may be used when necessary (e.g. before opening the next banking day). Account normalization can also be performed during approval of a contract or the corresponding Account Scheme.

2.1.2.15 Special Norm

The *Special Norm* field is used in indirect multi-currency normalization (see the section "Limit Normalization"). The field makes it possible to redefine the global parameter MULTICURRENCY_NORMALIZATION (see the document "WAY4™ Global Parameters"), in a specific Account Scheme.

When it is necessary to set up multi-currency account normalization using the *Special Norm* field, the following conditions must be taken into consideration:

- When the MULTICURRENCY_NORMALIZATION global parameter is set to one of the documented values, multi-currency normalization is performed according to the specified value of the global parameter regardless of the value of the *Special Norm* local parameter (including cases where the *Special Norm* field is left blank).
- The *Special Norm* local parameter only redefines the MULTICURRENCY_NORMALIZATION global parameter if the global parameter is set to an undocumented value, and:
- If the *Special Norm* field contains the "Default Multicurrency" value, or if the field is not filled in, multi-currency normalization for this Scheme's accounts can be performed either using standing payment orders (see the section "Multi-currency Normalization" of the document "Standing Payment Orders") or through indirect multi-currency normalization (see "Limit Normalization").
- If the *Special Norm* field contains the "None" value, multi-currency normalization is not performed.

> When performing multi-currency normalization using standing payment orders, the *Special Norm* parameter is not used.

2.1.2.16 Cr Lim Posting

This field allows for configuring how credit limits are posted and reflected in accounts of contracts that use this Account Scheme.

The field contains a drop-down list of possible values and depends on the global parameter CREDIT_LIMIT_POSTING (see the document "WAY4™ Global Parameters"):

- If the global parameter CREDIT_LIMIT_POSTING is set (the parameter value is "Y" or "N"), the *Cr Lim Posting* field of the Account Scheme is not analysed.
- If the global parameter CREDIT_LIMIT_POSTING is disabled (if the parameter is not set or its value is empty (NULL)), the mode for showing a credit limit in accounts can be enabled in the *Cr Lim Posting* field of the Account Scheme:
- When the value of the field is "Yes", credits limits are reflected in accounts based on this Account Scheme's templates.
- When the value of the field is "No", credit limits are not reflected in accounts of this Account Scheme.
- If the global parameter CREDIT_LIMIT_POSTING is not set, and the *Cr Lim Posting* field for the Account Scheme is not filled in, the *Cr Lim Posting* field for the financial institution is checked and its value is used.

- If the global parameter CREDIT_LIMIT_POSTING is not set, and the *Cr Lim Posting* field for the Account Scheme is not filled in, nor is it filled in for the financial institution, credit limits are not shown in this Account Scheme's accounts.

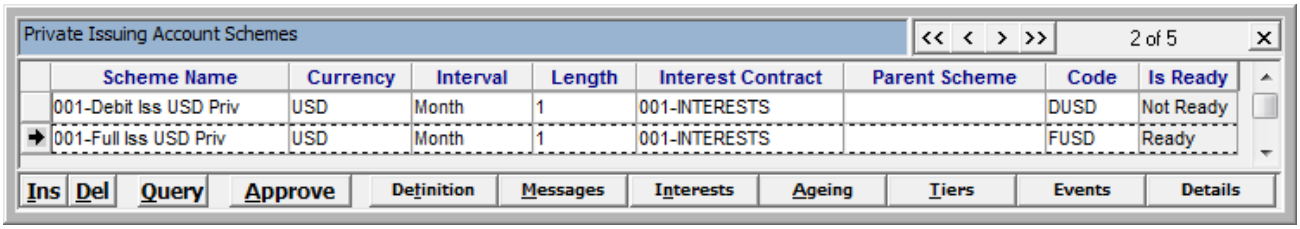
2.1.2.17 Special Parms

This field is used to specify additional parameters of Account Schemes as tags. See the section "Tags used when configuring Account Schemes and account templates" of the "Setup Tags" document.

2.2 Special Forms for Working with Account Schemes

In most cases, it makes sense to use special forms containing a list of Account Schemes selected by financial institution and contract category:

- To view and enter information about Account Schemes for private clients' issuing contracts, use the "Private Issuing Account Schemes" form (see figure) that is opened by the menu item "Full → Configuration Setup → Products → Issuing Private Products → Private Issuing Account Schemes".
- To view and edit information about Account Schemes for corporate clients' issuing contracts, use the "Corporate Issuing Account Schemes" form (Full → Configuration Setup → Products → Issuing Corporate Products → Corporate Issuing Account Schemes).
- To view and edit information about Account Schemes for acquiring contracts, use the "Acquiring Account Schemes" form (Full → Configuration Setup → Products → Acquiring Products → Acquiring Account Schemes).
- To view and edit information about Account Schemes for bank contracts, use the "Bank Account Schemes" form (Full → Configuration Setup → Accounting Setup → Bank Account Schemes).



Form for entering and editing Account Scheme data for private clients' issuing contracts

Note that sets of fields and buttons in the above forms may differ both from each other and from those used in the "Account Schemes" forms.

3 Configuring Account Templates

This section contains a description of parameters and features of account template setup.

3.1 Account Types

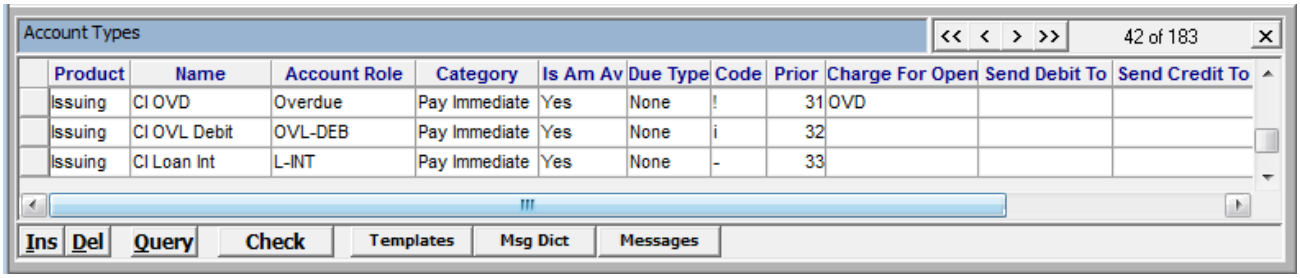
3.1.1 "Account Types" Form

An account template description in an Account Scheme contains its type (account type). The type of an account determines some of its properties.

The "Account Types" form (Full → Configuration Setup → Accounting Setup → Account Types) contains a list of account types.

> Starting with version 03.41.30, the standard account type "Cl Deposit" has been renamed to "CH Current/Credits".

This does not affect existing configurations and will only be reflected in new system installations (when WAY4 is first installed, beginning with version 03.41.30, the account type "Cl Deposit" will be absent from the "Account Types" form).



Form for registering and configuring account types

The "Account Types" form (see figure) contains the following fields:

- *Product* – drop-down list of Product categories (Issuing, Acquiring, Accounting, Bank Accounting).
- *Name* – account type name. The name must be unique within the same Product category (*Product* field).
- *Account Role* – drop-down list to select a role group. The field's list is created on the basis of the system handbook of role groups. For more information about role groups, see the section "Classification of Account Types (Account Roles)".
  In the "Account Types" form, several types of account can be created for one group (one role) in order to resolve technical tasks.
  Example.

Two account types ("Loan Retail" and "Loan Cash") with the "Loan" role are created for one Product. These account types are created to separate conditions for loan grace periods depending on the transaction type: when making retail transactions with a credit card a grace period is provided, and when withdrawing cash, no grace period is provided. These types of "Loan" group accounts have unique names (*Name* field) and codes (*Code* field).

- *Category* – drop-down list of account type categories; a category determines specifics of processing this type of accounts; the list of account categories is predefined in the system and contains the following values:
- "Pers Limit" – when lower limit normalization is performed for an account of this category, the threshold value will be the contract's credit limit, not the value of the *Low LimitAmount* field of the account template (see examples in section "Limit Normalization Mechanism"). This category is assigned to loan accounts.
- "Cr. Lim Payment Due" – this category is used to set up account templates for recording amounts made due as part of credit limit. In this case, when contract normalization is performed, balances of a loan account ("Cl Loan", category "Pers Limit") and a due account ("Cl Paym Due", category "Cr.Lim Payment Due") are totalled and compared with the contract's credit limit; therefore, when funds are transferred from the loan account to the due account, limit normalization (OVL → Cl Loan) is not performed for the loan account (see example 2 in the section "Limit Normalization Mechanism").
- "Cr. Lim Overdue" – this category is used in the same way as the "Cr. Lim Payment Due", but for overdue loan accounts ("Cl OVD").
- "Payment Due" – when a card statement is generated, balances of accounts with this category are totalled to show the due amount.
- "Pay Immediate" – when a card statement is generated, balances of accounts with this category are totalled to show the total due amount whose due date has already arrived.
- "Shared Limit" – this category is used as part of a special scheme of accounting. In this case, a credit limit is distributed among all contract accounts with the "Shared Limit" category. When loan interest is made due, balances of all "Shared Limit" accounts are totalled, and the amount from the account with the highest priority is made due. It is not recommended that accounts with the "Shared Limit" category be used together with accounts with "Pers Limit", "Cr. Lim Payment Due", and "Cr. Lim Overdue" categories. Accounts with the "Shared Limit" category do not support lower limit normalization.

  > Accounts with the "Shared Limit" category implement an obsolete type of account behavior. It is recommended that they only be used after a consultation with Customer Support.

- "Primary" – it is recommended that this category be assigned to deposit client accounts.

> This category should be used for accounts with a positive balance, on which interest is calculated in "On Request Only" mode (value of the account template's *Interest Algorithm* field; see the section "Interest Properties"). Otherwise, interest on a positive balance will not be accrued in this mode.

- "Dispute" – it is recommended that this category be assigned to dispute accounts.
- "Transit To" – category for transit accounts; groups of accounts with this category may be used in custom reports.
- "Other" – other accounts; assigning this category has no impact on account processing.
- *Is Am Av* – shows whether the account balance is considered during contract balance calculation:
- When the "Yes" value is set, the account balance is considered during balance calculation.
- When the "No" value is set, the account balance is not considered.
- *Due Type* – due normalization type; for more details on this field's values, see the section "Ageing". During account template setup, a due normalization type can only be changed to another type from the same group. The following groups are used in the system: "Payment Due", "None", "End Cycle Due", "Quarter", "Long Payment Due", "Sliding", "Value Date Due", "Sliding + Clear", and "Fixed Date Due". See the section "Ageing" for a description of *Due Type* field values.
- *Code* – account type code; the field is filled in using letters of the Latin or national alphabet, digits, or ASCII characters.
  An Account Scheme may not contain two account templates of the same type in the same currency. The system controls this limitation using the *Code* field. Therefore, account type codes should be unique at least in the same Product category. Note that the system differentiates between uppercase and lowercase characters in this field.

  > In the standard WAY4 configuration, a dispute account (*Category* = "Dispute") has code "D", and a deposit account (*Category* = "Primary"), code "P". It is recommended that values of the *Code* field for these account types not be changed since the above code values are used in some procedures.

- *Prior* – determines account priority in the Account Scheme. Priority values affect the order of interest accrual in accounts and repayment of loan account balances. When a new template is added to an Account Scheme, the default value from the account type table is specified in the field. The value can be redefined by users.
- *Charge for Open* – type of the fee charged when an account is opened; in this case, an account is considered opened when a non-zero balance appears in the account.

  > This field is left for backward compatibility. It is not recommended that it be used.

- Fields *Send Debit To* and *Send Credit To* are used in WAY4 to describe asset and liability account pairing: for a liability account, the *Send Debit To* field specifies an asset account that will be used

to debit the current account. Correspondingly, for an asset account, the *Send Credit To* field specifies a liability account that will be used to credit the current account.

The [Check] button is used after changing account type parameters. When this button is clicked, the parameters of the account type and account templates linked with this account type are compared. If the changed parameters of the account type are inherited by the account template, and the corresponding Account Scheme was not approved, an error message is shown after changes to the account type. The [Messages] button is used to get information about Account Schemes that must be approved.

The [Messages] button is used to analyse errors that occurred when comparing account type and account template parameters (see the description of the [Check] button). Clicking the [Messages] button opens the "Messages for..." form containing data on Account Schemes that must be approved. After a scheme is approved, changes to contracts must be applied using a separate procedure "Apply Account Scheme Changes" (Full → Configuration Setup → Products → Apply Account Scheme Changes) or by executing the "Contracts Daily Update" procedure (Full → Daily Procedures → Start of Day Step by Step → Contracts – Daily Update).

After changing account type parameters, it is mandatory to compare (Check) the parameters of the account type and account templates linked with this account type.

3.1.2 Classification of Account Types (Account Roles)

Account types are classified into groups depending on their use (depending on account role), for example "Deposit", "Loan", and "Loan Interest". These account role groups (Account Roles) are kept in a system handbook. The handbook can be viewed:

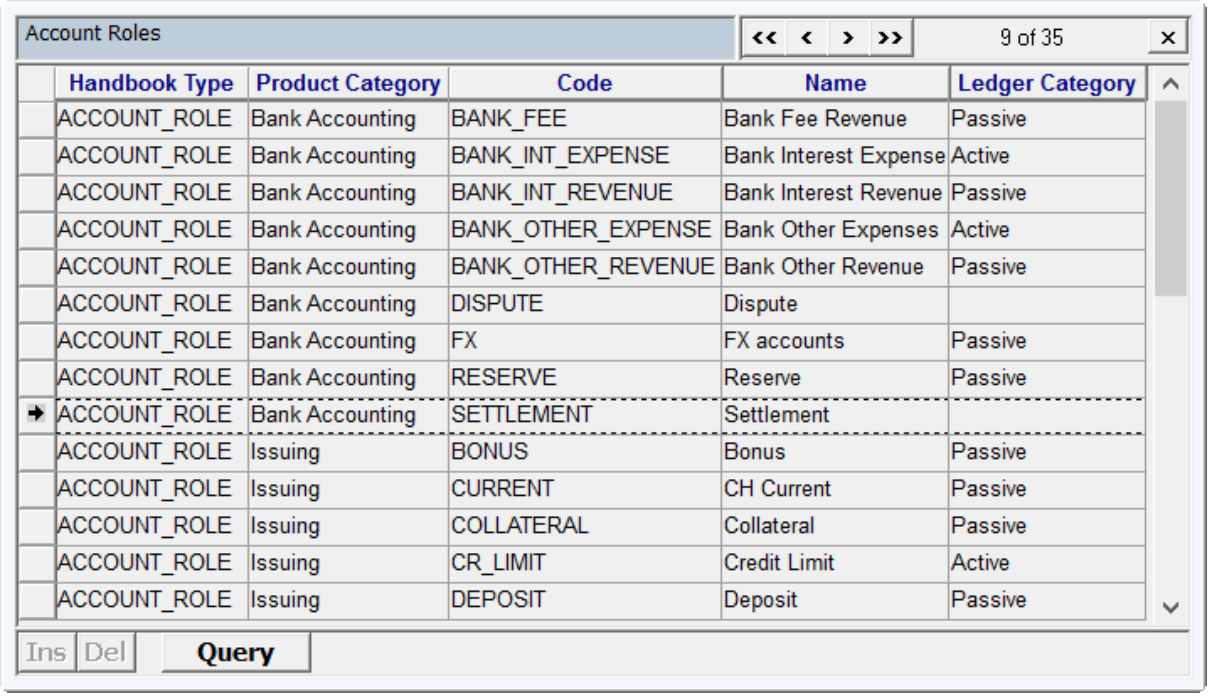
- In the "Standard Handbooks" form ("Full → Configuration Setup → Client Classifiers → Standard Handbooks", records with the "ACCOUNT_ROLE" code).
- In the "Account Roles" form (Full → Configuration Setup → Accounting Setup → Account Roles), see figure.

If new records must be added to the handbook, contact OpenWay.

Role groups are used to do the following:

- Arrange contract accounts for them to be clearly shown in the interface.
- Additionally classify financial information for contract accounts. Role groups make it possible to obtain non-overlapping balances for contract accounts and can be used to provide data on balances and turnover for contract accounts in reports and statements (for example, a description of operations in account statements, description of GL entries in reports). In the current WAY4 version, role groups are used when exporting data to the WAY4 Datamart module.

Starting from version 03.41.30, a new role group has been added – "CH Current". It is recommended to assign this role to accounts for a cardholder's own funds. The standard account type "Cl Deposit" has been renamed to "CH Current/Credits". This does not affect existing configurations and will only be reflected in new system installations (when WAY4 is first installed, beginning with version 03.41.30, the account type "Cl Deposit" will be absent from the "Account Types" form).



"Account Roles" form

The form contains the following fields:

- *Handbook Type* – handbook code. For account role group handbook records, the "ACCOUNT_ROLE" value is specified in this field.
- *Product Category* – determines the Product category to which an account group belongs.

  In the "Standard Handbooks" form, the *Filter 1* field specifies a particular Product category to which the account group belongs to:

  - "B" – Products for bank contracts ("Bank Accounting" category).
  - "M" – Products for acquiring contracts ("Acquiring" category).
  - "C" – Products for issuing contracts ("Issuing" category).

  An account type's role determines the Product category to which it belongs, therefore, this account should not be used in other Product categories (for example, a "Loan" type account should not be used in acquiring Products).

> There are several role groups that can be used in several Product categories. These are the groups "Dispute", "Credit Limit", "Full Credit Line", "Unused Credit Line", "Technical", and "Reserve" (see the table). For example, account types with the "Dispute" role (in the handbook, the "Dispute" role belongs to the "Bank Accounting" category) can be used in bank issuing and acquiring Products (Account Schemes). That is, the "Dispute" role with the "Bank Accounting" category (in the "Account Roles" handbook) can be set for an account type with the "Issuing" category (in the "Account Types" form). These exceptions from the general rule are because accounts of these groups (except the "Technical" account group) that are actually bank accounts, in a number of cases can be shown in client contracts. For example, in the case of individual portfolio reserving for loans included in one portfolio, a bank contract is registered for reserving from revenue/expense accounts (to generate/replenish reserves). The accrued reserves themselves will be shown in client loan contract accounts (for group portfolio reserves, accrued reserves will be shown in bank contract accounts).

- *Name* – account role group name.
- *Code* – account role group code.
- *Ledger Category* – account category (asset/liability account). This field is used when working with the WAY4 Datamart module.

Account role groups

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Use**
Credit and debit issuing products / Credit product			
Issuing	Deposit	Deposit account for recording a client's (other than merchant) own funds.	Current accounts of individuals and legal entities, special card accounts in debit and credit cards, deposit products
Bank Accounting	Credit Limit	Full credit limit approved for this credit agreement	Credit cards, non-card credit products These account types can be used in bank and client Account Schemes.

Product category in the "Account Roles" handbook	Role name	Description	Use
Bank Accounting	Unused Cr Line	Unused part of an approved credit limit	Credit cards, non-card credit products These account types can be used in bank and client Account Schemes This group is similar to the "Credit Limit" group, but put in a separate group to get non-overlapping balances on the role group level
Issuing	Loan	Loan debt	Credit cards, non-card credit products (including all technical parts of a loan other than overdue; for example, "Cash", "Retail", "Minimal Payment")
Issuing	Overdue	Overdue loan debt	Credit cards
Issuing	Over-limit	Over-limit debt (unauthorized overdraft)	Credit cards
Issuing	Fee	Fee (client obligation)	Credit cards, non-card credit products (for example, "Fees payable for less than 30 days", "Fees payable for more than 30 days")
Issuing	Overdue Fee	Overdue fees	Credit cards, non-card credit products
Issuing	Loan Interest	Loan interest (client obligation)	Credit cards, non-card credit products
Issuing	Overdue Interest	Overdue loan interest	Credit cards, non-card credit products
Issuing	Over-limit Interests	Over-limit interest	Credit cards, non-card credit products

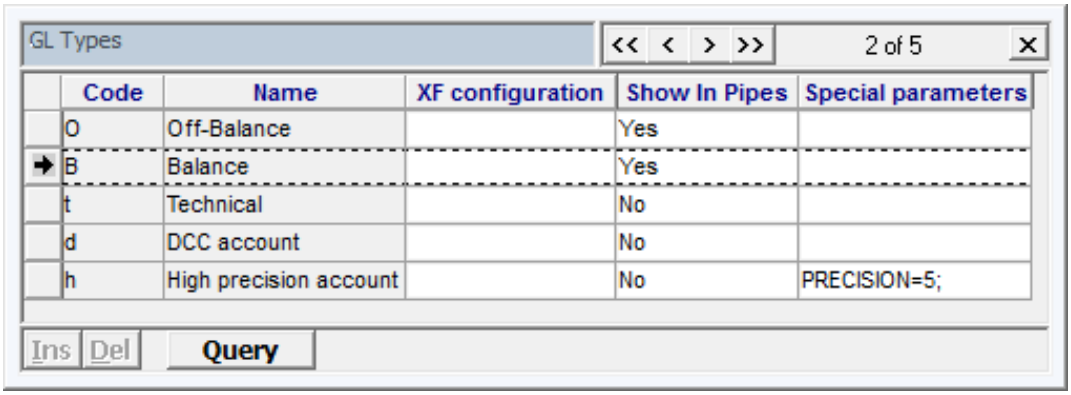
Product category in the "Account Roles" handbook	Role name	Description	Use
Issuing	Penalty	Fines and penalties	Credit cards, non-card credit products
Issuing	Guarantee	Guarantees, securities	Credit cards, non-card credit products (shows financial evaluation of third-person guarantees)
Issuing	Collateral	Collateral	Non-card credit products (shows monetary value of a collateral item (items), subject to re-evaluation due to depreciation)
Bank Accounting	Reserve	Generated reserves for a loan/ overdraft Revenue from replenishment of reserves. Bank expenses for generation of reserves	Credit cards, non-card credit products. All types of account created for recording various reserves belong to this group: "Reserve Loan", "Reserve OVL", "Reserve Interest", "Reserve OVD", "Reserve Fees", "Reserve Unused Credit Line". These account types can be used in bank (NNN-**RESERVE*) and client Account Schemes
Issuing	Bonus	Loyalty program bonus points	Loyalty Products
Deposit Products			
Issuing	Deposit	Account from recording current deposit amount (deposit)	Current accounts, special card accounts in debit and credit cards, deposit products
Issuing	Deposit Interest	Deposit interest (obligation to a client)	
Technical accounts			

Product category in the "Account Roles" handbook	Role name	Description	Use
Issuing	Technical	Technical account	Used in various Products, of any category to record funds in off-balance technical accounts (in these technical accounts, amounts are shown that are already considered in one or several main groups). For example, minimum balance for a deposit, initial amount of deposit, minimum payment on credit. Technical accounts are not assigned GL account numbers from an account plan.
Bank accounts (revenue/expense)			
Bank Accounting	Bank Fee Revenue	Revenue from issuing fees, managing client accounts, fines, penalties, for merchant acquiring.	Revenue from fees for issuing activity (bank contracts "NNN-CLIENT_FEE", "NNN-MERCHANT_FEE").
Bank Accounting	FX accounts	Revenue from positive FX difference. Expenses from negative FX difference	Bank contracts NNN-**FX*
Bank Accounting	Bank Interest Revenue	Revenue from credit interest	Bank contracts "NNN-**INTEREST*"
Bank Accounting	Bank Interest Expense	Expense for deposit interest	Bank contracts "NNN-**INTEREST*"
Bank Accounting	Bank Other Expense	Expenses for fees to sponsor banks, IPS, losses due to overdue debt and fraud	Bank contracts "NNN-**_NOSTRO
Bank accounts (Settlements)			

Product category in the "Account Roles" handbook	Role name	Description	Use
Bank Accounting	SETTLEMENT	Correspondent account (full or partial mirror of the account in the CBS – for example, entries for crediting a correspondent account can be made in the CBS and not reflected in WAY4).	Settlement with a payment system, agent banks For example, "Nostro – correspondent account", "Nostro Suspense", "Incoming Suspense", "Outgoing Suspense", "Settlement Fees" (only for IPS), "Settlement Expenses/Revenue")
		Bank accounts (Cash)	
Acquiring	Cash/Teller	Cash	ATM accounts, Cash-in devices, virtual teller contracts, Main contracts of cash POS terminals.
		Acquiring (Retail)	
Acquiring	Merchant, Merchant Receivable, Merchant Current	Obligations to merchants. Merchant own funds	1. Merchant contract accounts (funds unavailable to a merchant). Used to withhold settlements with merchants – accounts "Merchant Receivable" and "Merchant Current" (if they have the same number). 2. Merchant contract accounts, if they are managed in WAY4.
		Dispute accounts	
Acquiring Issuing	Dispute	Account for reflecting disputed amounts in client schemes, payment system schemes, inter-branch settlements.	Dispute operations until resolution.

3.2 Configuring the "GL Types" Form

Various sections for accounting are configured in the "GL Types" form (Full → Configuration Setup → Accounting Setup → GL Types), see figure. Settings in this form for a certain section are used for all account templates of the specified type. Settings can be redefined for a specific account template.



"GL Types" form

The form contains the following fields:

- In the *XF Configuration* field, XF accounts are configured with the tags OWN_XF_ACCOUNT;, OWN_ALT_XF_ACCOUNT;, ALT_XF_ACCOUNT;, XF_CONTRACT, XF=< >;. The tags are described in the section "Configuring High Precision Account Templates". Tags can be redefined in account templates (see the section "Off-Balance Accounting Subsystem").
  The tag XF_<code of the corresponding accounting section>=<>; can be specified in the *XF Configuration* field. This tag makes it possible to configure different XF accounts for correspondence with different sections for accounting (the tag value is configured in the same way as the XF tag, see the section"Off-Balance Accounting Subsystem"). This tag cannot be redefined in an account template.

  > XF accounts are determined as follows:
  >
  > - Configuration in an account template.
  > - The tag XF_<code of the corresponding accounting section>=<>; in the "GL Type" form.
  > - XF=< >; in the "GL Type" form.

- *Show in pipe* – this field is used to configure export of account entries (this setting makes it possible to hide technical entries):
- "Y" – entries with this GL Type will be exported using the GL Transfers Export pipe (on condition of the appropriate setting for the HIDE_BY_GL_TYPE pipe parameter).
- "N" – entries with this GL Type will not be exported using the GL Transfers Export pipe (on condition of the appropriate setting of the HIDE_BY_GL_TYPE pipe parameter).

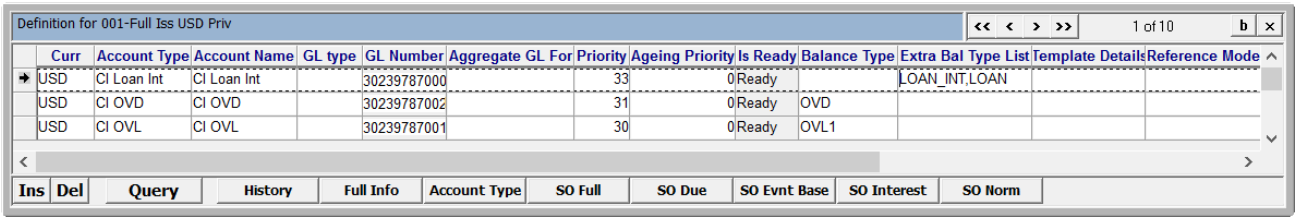
  > To filter subsidiary GL entries by the GL Type parameter, the corresponding value of the GL Type parameter must be set in the account template.

See the document "Export of Subsidiary GL Entries and GL Entries in UFX Format".

- By default, the tag PRECISION=6; is specified in the *Special Parameters* field. This tag determines the number of decimal places that will be used by default for "High Precision" accounts (if this tag is not set in account templates). See the section "High Precision Accounting".

3.3 Full Information about Account Scheme Templates

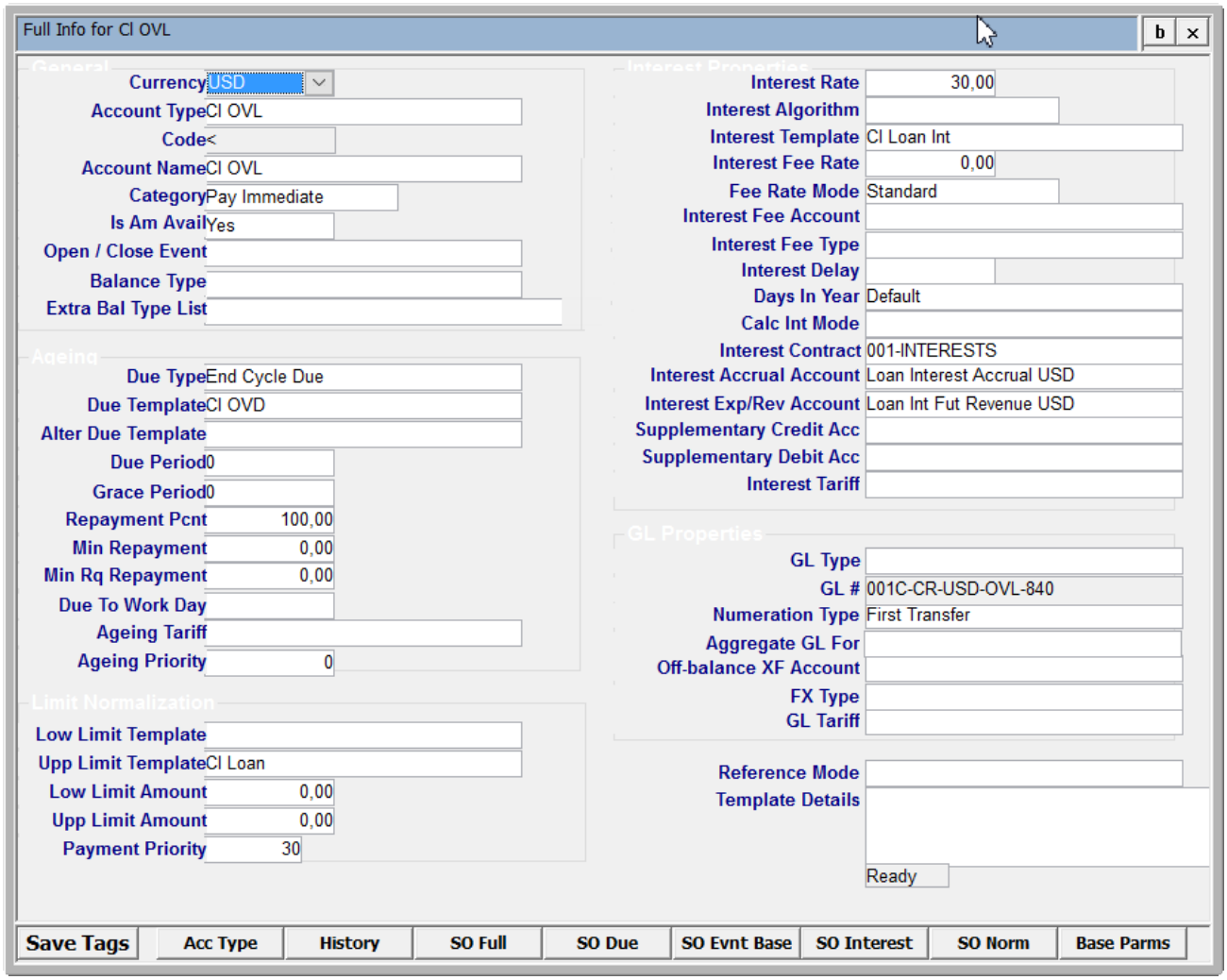
For more detailed information about each template in an Account Scheme, click the [Definition] button in the "Account Schemes" form (see the figure in the section ""Account Schemes" Form) or in a special form for working with Account Schemes (see the figure in the section "Special Forms for Working with Account Schemes"). As a result, the "Definition for <name of Account Scheme>" form will open. See an example of the form for private clients' issuing contracts in figure.



Form for accessing account templates

In the "Definition for <name of Account Scheme>" form that opens, click the [Full Info] button. As a result, the "Full Info for <name of account template>" form will open. This is the most convenient form for registering new account templates or modifying old ones (see figure).

Fields used in special forms for working with Account Schemes of various contract categories (see the section "Special Forms for Working with Account Schemes") are the same as the ones used in the "Full Info for <name of account template>" form. An exception is the form for entering and editing Account Scheme data for bank contracts (Full → Configuration Setup → Accounting Setup → Bank Account Schemes). The form contains the *Fin Institution* and *HeadOffice GL#* fields that are used to set up interbranch routing (see the"Interbranch Transactions" section in the document "Financial Institutions") and absent from the account template full information form.



Form for viewing and entering information about Account Scheme accounts

To access information about an account template type, click the [Acc Type] button. Account type parameters are registered and changed in the "Account Types" form (Full → Configuration Setup → Accounting Setup → Account Types).

The [History] button is used to open the form containing account template modification history.

Buttons [SO Full], [SO Due], [SO Evnt Base], [SO Interest], and [SO Norm] are used to define and set up standing payment orders for an account template. For more information about standing payment orders, see the document "Standing Payment Orders".

The [Base Parms] button opens a form containing the editable parameters *Charge for Open* (account type parameter), *Is Am Available*, *Category*, and parameters *Account Type*, *Account Name*, and *Currency* that can be viewed.

The [Tagged Data] button is used to optimize work with the *Template Details* field. Clicking this button opens the grid form for entering and editing tags in the *Template Details* field (see the section "Entering and Editing Tags").

Fields in the "Full Info for <name of account template>" form are divided into five groups:

- "General" – template's main parameters.
- "Ageing" – due normalization parameters.

- "Limit Normalization" – amount normalization parameters.
- "Interest Properties" – interest accrual parameters.
- "GL Properties" – GL parameters.

3.3.1 General

This group of fields in the "Full Info for <name of account template>" form (see the figure in the section "Full Information about Account Scheme Templates") is used to enter main account information.

- *Currency* – template's account currency.
- *Account Type* – template's account type.

  > An Account Scheme may not contain two account templates of the same type in the same currency. If it is necessary to add an account template whose properties are the same as the properties of another account template in the Account Scheme, add a new account type (see the section "Account Types").

- *Account Name* – account name.
- *Category* – account type category.
- *Is Am Avail* – shows whether the account balance is considered during contract balance calculation.

  > It is not recommended to use accounts with the "Yes" value in the *Is Am Avail* field in bank and acquiring contract Account Schemes.
  >
  > When making transactions in an account with this attribute, the contract is blocked for updating available amounts. When processing a large number of transactions (as in the case of bank contracts and acquiring contracts), this may lead to system failure.

  If a balance type (see the description of the *Balance Type* field) is set for an account with "No" in the *Is Am Avail* parameter, when a transaction in the account is made, the amount available is calculated based on this balance type's value.
- *Open/Close Event* – name of the Event type that is opened when funds appear in the account and closed when the account balance becomes zero.

  > An exception is the situation when during processing of a macrotransaction leading to a certain Event being closed because the account balance becomes zero, another Event is opened for the account due to balances in another account/balance (an Event related to an account template and/or balance type). These Events can be not linked (for example, using the Event package). In this case, the Event will be closed if the Event type contains the NOT_USED_IN_CHAIN tag. Otherwise, the Event will not be closed.

- *Balance Type* – drop-down list of balance types; balances of accounts generated using this template or, depending on the balance type parameters, account turnovers over the billing cycle will be considered when the value of the selected balance type is calculated (see section "Registering Balance Types" of the document "Balance Types").
- *Extra Bal Type List* – if it is necessary to use the account balance or turnover in several balance types, list the codes of the balance types separated by commas in the *Extra Bal Type List* field.

> Balances must be recalculated if changes are made in the *Balance Type and Extra Bal Type List* fields. For more information, see the section "Recalculating Balance Type Values" of the document "Balance Types".

3.3.2 Ageing

This group of fields in the "Full Info for <name of account template>" form (see the figure in the section "Full Information about Account Scheme Templates") is used to determine rules for funds transfer during due normalization.

When a banking day is opened (see the section "Start of Day Procedure" in the document "Daily Procedures"), the contract processing procedure (Contracts Daily Update) is executed after setting a new banking date. Among other tasks, the procedure transfers funds between contract accounts according to preconfigured due normalization rules.

- Field *Due Type* – due normalization type; the value of this field determines how the due date is calculated:
  - "Value Date Due" – the normalization date is calculated by adding to the date when funds are transferred to the account the number of banking days determined by the *Value Days* parameter of the Service (or by a tariff with the "Service Value Days" role set in the Service) used for transaction posting (see the "Posting" section in the document "WAY4™ Service Packages"). If the *Value Days* field is not filled in for a Service (equal to zero) and the "Service Value Days" tariff is not set, the *Due Period* field of the account template is used (or the value of a tariff with the "Ageing" role set in the template).Funds are transferred for the amount of transactions that were made on the date funds were credited to the account (that is, the normalization amount consists of the amount of funds flow in the account for the corresponding banking day). The system differentiates between business days and weekends using a business calendar (see the section "Business Calendar" of the document "WAY4™ Dictionaries").
  - "End Cycle Due" – the normalization period is determined by the end of the the corresponding billing cycle, depending on the value of the *Due Period* field ("0" – normalization at the end of the current billing cycle; "1" – normalization at the end of each second billing cycle from the contract opening date; "2" – normalization at the end of each third billing cycle from the contract opening date, etc.). The contract opening date is considered to be the date from the contract's *Open/Close* (DATE_OPEN) field. The current number of the billing cycle can be seen in the "Accounts for <contract name>" form in the *N Of Cycle* field. The normalization amount is the account's incoming balance on the

normalization date; macrotransaction posting is affected by the value of the POST_DUE global parameter (see the document "WAY4™ Global Parameters").

- "Contract Due" – the normalization date is determined based on calculated contract functional dates. Any functional date can be used to post due normalization. The USE_DUE_DATE=<functional date code>; tag is used to do so (for example, USE_DUE_DATE=LP_DATE;). By default, "Delinquency Date" (DLQ_DATE) is used to post due normalization, or "Due Date" (DUE_DATE) if DLQ_DATE is not set. For more information about functional dates, see the document "Contract Functional Dates".
  If a calculated date falls on a weekend/holiday according to the financial institution's (or Account Scheme's) calendar, and for the financial institution every day, including weekends/holidays is opened when the "Contracts – Daily Update" procedure is performed, due normalization will be performed when opening/closing (according to the global parameter PAYMENT_DUE_ADVANCE or the same tag in the Account Scheme) the first working day after the calculated date, according to the calendar of the financial institution/Account Scheme.
  Funds are transferred for the amount of funds flow in the account that was accumulated for the period from the start date of the billing cycle to the normalization date.
  When a transaction is made in the period between the normalization date and the start date of the next billing cycle, normalization is performed immediately. This is not a standard situation, since in general, the "Contract Due" normalization type is used for accounts to which funds are transferred form accounts with the "End Cycle Due" normalization type. Tat is, funds from a "Contract Due" account are transferred on a calculated normalization date and before the start of the next billing cycle, there is generally no flows in these accounts.

When the *Due Type* field contains the "Contract Due" value, the value of the DUE_TO_WRK_DAY parameter is always interpreted by WAY4 as "Y" (regardless of the global parameter's settings), and can only be redefined in the date scheme (see the section "Configuring Rules for Calculating Functional Dates" of the document "Contract Functional Dates").

- "Payment Due" – the normalization date is calculated by adding to the beginning of a billing cycle the number of calendar days specified in the *Due Period* field. The normalization amount is the account's incoming balance on the normalization date; macrotransaction posting is affected by the value of the PAYMENT_DUE_ADVANCE global parameter (see the document "WAY4™ Global Parameters"). If date schemes are used to define contract functional dates, it is recommended to use the "Contract Due" normalization type instead of "Payment Due" when configuring normalization.
- "Quarter" – the normalization date is determined by the number of quarters specified in the *Due Period* field ("1" – every quarter on the calendar date starting a quarter; "2" – twice a year on the calendar date starting a half-year; "4" – once a year on the calendar date starting a year). The normalization amount is the account's incoming balance on the normalization date.

> The "Quarter" value should only be used for configurations in which the end of a billing cycle corresponds to the end of the quarter. For example, if the billing cycle corresponds to a calendar month, or to a quarter (the billing cycle is equal to three months), and there is no shift in the billing cycle (see the description of the *Date Type* field).
>
> If a billing cycle is measured in different units, or if it is shifted, when the value is "Quarter", due normalization will be performed when the first billing cycle is opened in the new quarter/six-month period/year (according to the value of the *Due Period* field). This means that when a billing cycle is shifted (for example, when the *Date Type* field value is "From Open Date"), normalization may be performed not on the first of the month, but at the end.

- "Long Payment Due" – this value is left for backward compatibility. Works in the same way as the "Payment Due" value.
- "Sliding" – used to set a normalization period (in calendar days) when the normalization date is calculated by adding the normalization period (the value of the *Due Period* field) to the date when funds are transferred to the account.
  Funds are transferred for the amount of transactions that were made on the day funds were credited to the account (that is, the normalization amount consists of the amount of funds flow in the account for the corresponding banking day).
  In general, the "Sliding" normalization type is used for accounts to which funds are transferred from accounts with the "End Cycle Due" normalization type. That is, funds from a "Sliding" account are transferred on a calculated normalization date and before the start of the next billing cycle, there is generally no flows in these accounts.
  Example 1:
  An Account Scheme contains templates "Cl Loan" and "Cl OVD". A loan must be repaid within 45 days after the transaction date: *Due Type* of the "Cl Loan" account is set to "Sliding", *Due Period* is 45 days.
  - On 01 March, the balance of the contract and all its accounts was zero.
  - On 05 March, a card transaction for 40 euro was made. As a result, the balance of the "Cl Loan" account became 40 euro.
  - On 05 April, a card transaction for 60 euro was made. As a result, the balance of the "Cl Loan" account became 100 euro.
  - On 19 April, the amount of 40 euro, which is the amount due for the transaction madde on 05 March, was transferred to the overdue loan account "Cl OVD". As a result, the balance of the "Cl Loan" became 60 euro, and the balance of the "Cl OVD account became 40 euro.
  - On 20 May, the amount of 60 euro, which is the amount due for the transaction made on 05 April, was transferred to the overdue loan account "Cl OVD". As a result, the balance of the "Cl Loan" became zero, and the balance of the "Cl OVD account became 100 euro.

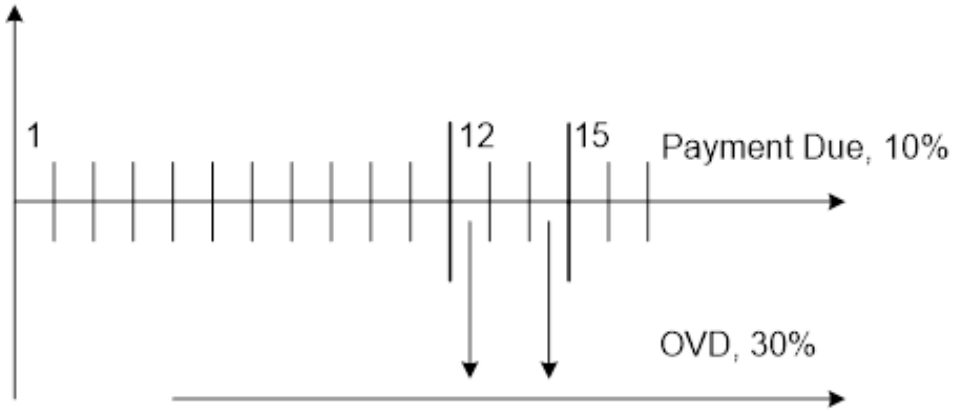
Example 2:
An Account Scheme contains templates "Cl Loan" and "Cl OVD". A loan must be repaid within 45 days after the transaction date: *Due Type* of the "Cl Loan" account is set to "Sliding", *Due Period* is 45 days.

- On 01 March, the balance of the contract and all its accounts was zero.
- On 05 March, a card transaction for 40 euro was made. As a result, the balance of the "Cl Loan" account became 40 euro.
- On 06 March, a card transaction for 60 euro was made. As a result, the balance of the "Cl Loan" account became 100 euro.
- On 10 March, the amount of 50 euro was credited to the card. As a result, the balance of the "Cl Loan" account became 50 euro. This payment fully repaid the amount due for the transaction made on 05 March and partially repaid the amount due for the transaction made on 06 March.
- On 20 April, that is 45 days after 06 March, the unpaid part of the amount due for the transaction made on 06 March (50 euro) will be transferred to the overdue loan account "Cl OVD".

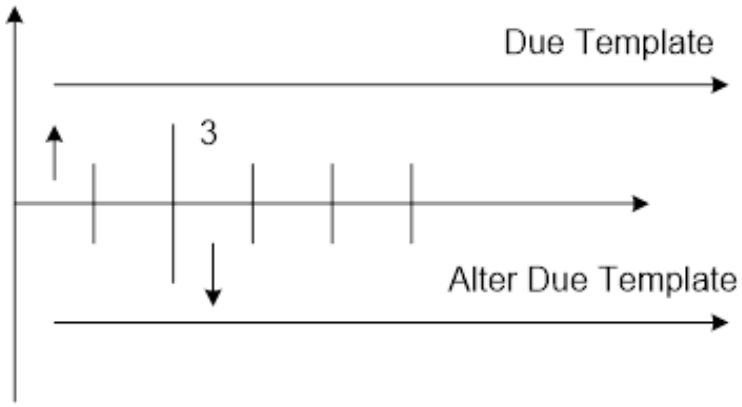
- "Sliding+Clear" – the normalization date is calculated by adding the normalization period (the value of the *Due Period* field) to the date of the first transaction that made the balance of the account non-zero. The normalization amount is the account's incoming balance on the normalization date.
- "Fixed Day Due" –the normalization date is the calendar day after the day of the month specified in the *Due Period* field.
  Funds are transferred for the amount of funds flow in the account that was accumulated for the period from the start date of the billing cycle to the normalization date.
  When a transaction is made in the period between the normalization date and the start date of the next billing cycle, normalization is performed immediately. This is not a standard situation, since in general, the "Fixed Day Due" normalization type is used for accounts to which funds are transferred form accounts with the "End Cycle Due" normalization type. That is, funds from a "Fixed Day Due" account are transferred on a calculated normalization date and before the start of the next billing cycle, there is generally no flows in these accounts.

- The *Due Template* field specifies the account to which funds are transferred from this account after the specified time period expires.
- *Alter Due Template* – if the *Alter Due Template* field is filled in, funds included in *Repayment* are transferred to *Due Template*, and the remaining funds are transferred to *Alte Due Template*.
- *Due Period* – the value of this field is used considering the normalization type (the value of the *Due Type* field) to determine the normalization period, that is the time period after which funds will be transferred from this account to the account specified in the *Due Template* field.
- When "Contract Due" is specified in the *Due Type* field, the *Due Period* field is not used to define normalization dates. In this case, a normalization date is set using contract functional dates and the value of the *Due Period* field is ignored.
- *Grace Period* – there are three ways to use this field:

- In account templates with value "Payment Due", "Value Date Due" or "Contract Due" in the *Due Type* field, the value in the *Grace Period* field means a loan's grace period in calendar days. This is usually used in due accounts ("Loan Payment Due"). The *Grace Period* value is subtracted from the *Due Period* value.
  For instance, assume that *Due Period* is "15", *Grace Period* is "4", and funds were credited to the account on the first day of the month. At the end of the *Due Period*, funds are transferred from the due account to the overdue account. The loan interest rate in the due loan account is 10%; the loan interest rate in the overdue account is 30%.



  If the client repays the loan by day 12 (15-4+1=12), loan interest at the rate of 10% will be accrued for this period. If the client repays the loan between day 12 and day 14, loan interest will be accrued at the rate of 10% as well (that is, at a preferential rate). If the client repays the loan on day 15 or later, loan interest will be accrued as follows: for the period up to day 12, at the rate of 10%, and for the remaining period, at the rate of 30%. Technically, the *Posting Date* of the entry between the due account and the overdue account will be day 12, and its *GL Date* will be day 15.
- In account templates with the "End Cycle Due" value in the *Due Type* field, the value in the *Grace Period* field means the delay before transferring funds to the account specified in the *Alter Due Template* field.



  Example: *Repayment Percent* is "40", and both *Due Template* and *Alter Due Template* are specified. *Grace Period* is "3". In this case, 40% of the amount will be transferred to *Due Template* at the end of a billing cycle, and the remaining 60% of the amount will be transferred to *Alter Due Template* in *Grace Period* calendar days.

    - If an account template contains the "Begin Balance" value in the *Interest Algorithm* field, the value in the *Grace Period* field is a shift from the first day of a billing cycle (in calendar days). The account balance for this date will be used for interest accrual. For more details, see the section "Interest Properties".
- *Repayment Pcnt* – amount percentage that must be transferred to the account specified in the *Due Template* field. The value "0" is an exception and means the same as the value "100" – that the entire amount is transferred. If the *Alter Due Template* field is filled in, the remaining amount will be transferred to this account. *Repayment Pcnt* is usually used to specify a loan percentage that must be transferred to the due account ("Cl Paym Due").
- *Min Repayment* – if the account balance (or the account balance multiplied by the value of the *Repayment Pcnt* field) is less that the value specified in the *Min Repayment* field, funds are not transferred to the account specified in the *Due Template* field. If a negative value is specified in the *MinRepayment* field, the system will use the opposite (positive) value as the value of the *MinRq Repayment* field. This is only applied when the *Min Rq Repayment* field is left blank.
- *Min Rq Repayment* – minimum amount that must be transferred to the account specified in the *Due Template* field of the account template:
    - If the account balance is less than the value specified in the*Min Rq Repayment* field, the entire account balance will be transferred to the account specified in the *Due Template* field.
    - If the account balance multiplied by the value of the *Repayment Pcnt* field is less than the value specified in the *Min Rq Repayment* field, the amount equal to the value of the *Due Template* field will be transferred to the account specified in the *Min Rq Repayment* field.
- *Due To Work Day* – redefines the global parameter DUE_TO_WRK_DAY affecting the process of posting due normalization macrotransactions to accounts (the parameter's main purpose is to shift the normalization date to a working day if the due date falls on a weekend or holiday). For more information, see the section "DUE_TO_WRK_DAY" in the document "WAY4™ Global Parameters". This parameter does not affect the procedure for determining the normalization date when the normalization type is "End Cycle Due" or "Quarter".
- *Ageing Tariff* – drop-down list of tariff types with the "Ageing" role registered in WAY4. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module. See the section "Tariffs with the "Ageing" Role" of the document "WAY4™ Advanced Tariff Management". If the *Due Type* field's value is "Contract Due", the *Ageing Tariff* field is not filled in. In this case, a normalization date is set using contract functional dates and cannot be redefined with an "Ageing" tariff. The Advanced Tariff Management module is not included in the WAY4 basic configuration and is delivered according to an additional agreement with OpenWay.
- *Ageing Priority* – this field affects the due normalization process for "End Cycle Due" and "Quarter" account types (these are Loan current accounts from which funds at the end of a cycle become due). First, due normalization is performed for accounts with a higher priority. The value of the *Payment Priority* field is considered. That is, if there are two accounts: Account 1 with value "3" in the *Ageing Priority* field, value "2" in the *Payment Priority* field and Account 2 with value "3" in the *Ageing Priority* field, value "1" in the *Payment Priority* fields, due normalization is first performed for Account 1.

3.3.3 Limit Normalization

> It is not recommended to configure limit normalization for Account Schemes of bank contracts and acquiring contracts. When making transactions in an account with this attribute, the contract is blocked for updating available amounts. When processing a large number of transactions (as in the case of bank contracts and acquiring contracts), this may lead to system failure.

3.3.3.1 Description of Fields

When balances of contract accounts change, it may be necessary to perform their normalization.

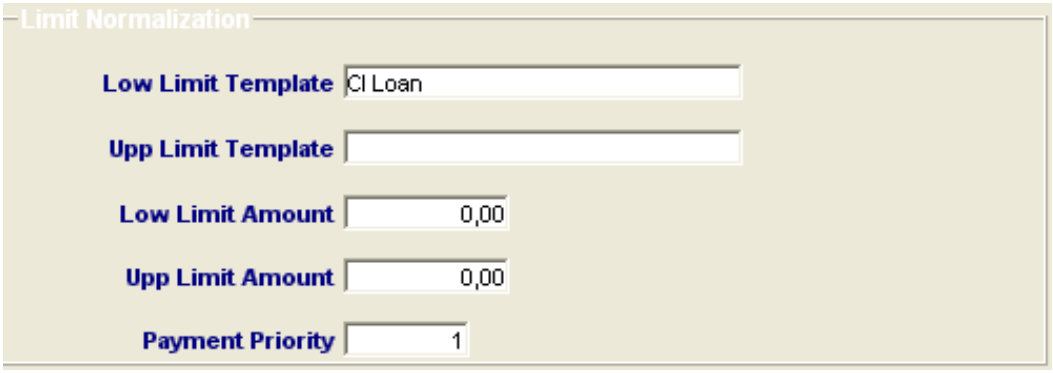
Balance limits are specified for each contract account. A related account to which overlimit or underlimit amounts must be transferred is specified for each limit. These parameters are set up in the "Limit Normalization" group of fields in the "Full Info for <name of account template>" form (see figure):

- *Low Limit Template* – account to which funds are transferred if the balance is lower than the specified limit.
- *Upp Limit Template* – account to which funds are transferred if the balance is higher than the specified limit.
- *Low Limit Amount* – lower limit; if the account balance is lower than the specified value, the funds are automatically transferred to the account specified in the *Low Limit Template* field.
- *Upp Limit Amount* – upper limit; if the account balance is higher than the specified value, the funds are automatically transferred to the account specified in the *Upp Limit Template* field.
- *Payment Priority* – account priority within the Account Scheme. Priority values affect the order of interest accrual in accounts and repayment of loan account balances. When a new template is added to an Account Scheme, a default value from the account type table is specified in the field. The value can be redefined by users.

  > If "0" is set in the *Payment Priority* field, when an entry for this account is made, limit normalization will not be performed.

  > It is not recommended to set a priority for account templates of bank contracts and acquiring contracts. When making transactions in an account with this attribute, the contract is blocked for updating available amounts. When processing a large number of transactions (as in the case of bank contracts and acquiring contracts), this may lead to system failure.

  The *Payment Priority* field is checked for the account template specified in the *Low Limit Template* or *Upp Limit Template* field. For this template, "0" cannot be set in this field.



Setup parameters of limit normalization for a deposit account

> If the "ONLINE_NORMALIZATION" global parameter (Full → Configuration Setup → Main Tables → Additional Global Parameters) is set to the "Y" value, normalization of accounts affected by a macrotransaction will be performed immediately during macrotransaction processing (see the document "WAY4™ Global Parameters").

To approve changes made in a template, click the [Approve] button in the "Account Schemes" form (see the figure in section ""Account Schemes" Form").

3.3.3.2 Limit Normalization Mechanism

The limit normalization mechanism is as follows:

- If an account balance is lower than the limit specified in the *Low Limit Amount* field, the difference is transferred to the account specified in the *Low Limit Template* field.

  > For accounts of the "Pers Limit" category (see a description of the *Category* field in section "Account Types"), a threshold value is the contract's credit limit, not the value of the *Low Limit Amount* field.

  Example 1.

- A contract's credit limit is 1,000 US dollars. The balance of the loan account ("Cl Loan") is 1,000 US dollars.
- The "Pers Limit" category is assigned to the loan account ("Cl Loan"), and the "Payment Due" category is assigned to the due account ("Cl Paym Due").
- At the end of a billing cycle, 10% of the "Cl Loan" balance is transferred to the due account ("Cl Paym Due") according to due normalization rules. Therefore, the balance of the "Cl Loan" account becomes 900 US dollars, and the balance of the "Cl Paym Due" account becomes 100 US dollars.

  Example 2.

- A contract's credit limit is 1,000 US dollars. The balance of the loan account ("Cl Loan") is 900 US dollars.
- A card transaction is maded for the amount of 100 US dollars.

- With this setup, the balance of the "Cl Loan" account does not exceed the contract's credit limit amount (900<1,000); as a result, the card transaction is reflected in the "Cl Loan" account (the balance of the "Cl Loan" account becomes 1000 US dollars), lower limit normalization is not performed for the "Cl Loan" account.

  Example 3.

- A contract's credit limit is 1,000 US dollars. The balance of the loan account ("Cl Loan") is 1,000 US dollars.
- The "Pers Limit" category is assigned to the loan account ("Cl Loan"), and the "CrLim Payment Due" category is assigned to the due account ("Cl Paym Due").
- At the end of a billing cycle, 10% of the "Cl Loan" balance is transferred to the due account ("Cl Paym Due") according to due normalization rules. Therefore, the balance of the "Cl Loan" account becomes 900 US dollars, and the balance of the "Cl Paym Due" account becomes 100 US dollars.
- A card transaction is maded for the amount of 100 US dollars.
- With this setup, the contract's credit limit is calculated as a sum total of balances of "Cl Loan" and "Cl Paym Due" accounts (900+100=1,000); as a result, lower limit normalization is performed for the "Cl Loan" account, and the card transaction is reflected in the "Cl OVL" account (the balance of the "Cl OVL" account becomes 100 US dollars).

- If an account balance exceeds the value of the *Upp Limit Amount* field, the difference is transferred to the account specified in the *Upp Limit Template* field.
- If an account balance is lower than the value of the *Upp Limit Amount* field, and the account specified in the *Upp Limit Template* field has a positive balance, an amount equal to the difference between the *Upp Limit Amount* and the account balance (but not exceeding the balance of the *Upp Limit Template* account) is transferred from the *Upp Limit Template* account to this account. For example, accrued loan interest is repaid in this way in the WAY4 standard credit scheme.

Another type of limit normalization is indirect multi-currency normalization. This mechanism is used if an Account Scheme contains accounts in different currencies.

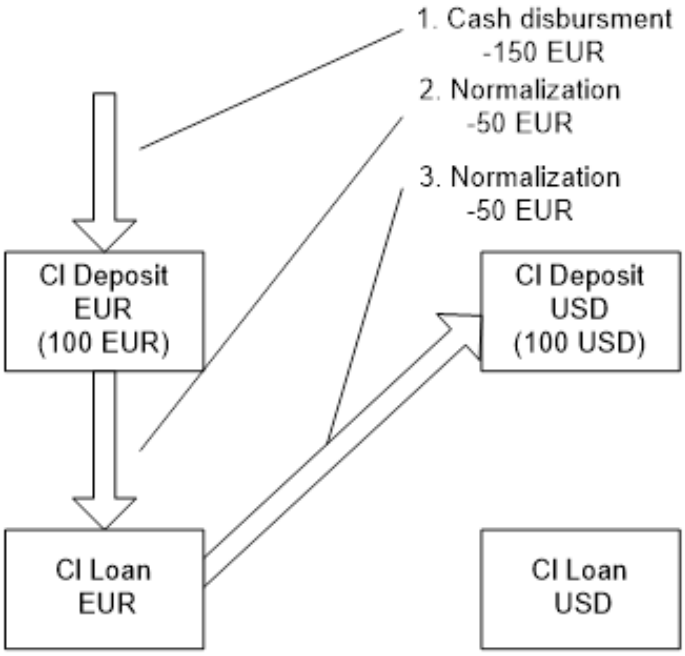
The standard mechanism of amount normalization between accounts in different currencies uses standing payment orders of a special type (see the section "Multi-currency Normalization" of the document "Standing Payment Orders").

The order in which normalization for an account is performed when there are payment orders is regulated by the order's *Priority* field (see the "Transaction Descriptions" section of the document "Standing Payment Orders").

Indirect multi-currency normalization (see figure) is enabled when the MULTICURRENCY_NORMALIZATION global parameter is set to "Y" (see the document "WAY4™ Global Parameters").

Indirect multi-currency normalization is possible in Account Schemes with more than two currencies. For example, an Account Scheme may use currencies EUR, USD and JPY. In this case, all three accounts participate in normalization, but the order in which their funds will be used to repay a loan is not regulated (when standing payment orders are used, the order is determined by order parameters).

For example, an Account Scheme whose main currency is EUR contains two accounts – Cl Deposit (USD) and Cl Deposit (EUR).



Multi-currency normalization

In case of indirect multi-currency normalization (see figure), when a transaction is performed, the amount is first debited from the "Cl Deposit (EUR)" account. If the transaction amount exceeds the limit specified in the *Low Limit Amount* field, the difference is transferred to the account specified in the *Low Limit Template* field. Then, part of the lacking amount will be debited to the "Cl Deposit (USD)" account.

This happens because when the MULTICURRENCY_NORMALIZATION global parameter is set to "Y", limit normalization of an account also affects the contract's accounts of the same type registered in other currencies.

During multi-currency normalization, the debited amount including the fee and rounding may not exceed the amount of the balance in the account being debited.

3.3.4 Interest Properties

This group of fields of the "Full Info for <name of account template>" form (see the figure in the section "Full Information about Account Scheme Templates") is used to specify interest accrual conditions.

- *Interest Rate* – interest rate. When it is necessary to change the interest rate for a template based on which accounts have already been opened in the database and transaction activity has already been registered in the accounts, it is recommended that users proceed as follows:
  - The interest rate is changed during the banking day preceding that on which the rate will become effective.
  - After changing the interest rate in the account template, approve the changes by clicking the [Apply] button in the Account Scheme form.

- The new interest rate will be used after running the "Apply Account Scheme Changes" procedure within the Contracts Daily Update procedure, meaning it will become effective the day after the changes were made. The "Contracts Daily Update" procedure is run automatically within the procedure for opening a new banking day.

> The "Apply Account Scheme Changes" procedure may take significant time. To reduce the time of the Contract Daily Update procedure, the "Apply Account Scheme Changes" procedure can be run separately using the menu item "Full → Configuration Setup → Products → Apply Account Scheme Changes". This procedure should be run after processing the last macrotransaction for the current day.

- *Interest Algorithm* – interest accrual algorithm. For more details on the values of this field, see the section "Interest Algorithm" of the document "Interest Accrual".
- *Interest Template* – account to which interest is accrued.
- *Interest Fee Rate* – rate of the fee charged in the following cases:
- Charging a fee from account interest income; used for automatic allocation of income tax on deposit interest.
- Accruing penalties.
- For more information, see the section "Interest Fee Rate" of the document "Interest Accrual".

> Previously, the *Interest Fee Rate* field was used as a basic functionality for setup parameters to levy an income tax received as financial benefit. Currently, it is recommended that charge of this tax be set up using the WAY4 Tariffs module. The module is supplied according to an additional agreement with OpenWay.

- *Fee Rate Mode* – drop-down list of fee charge algorithms. For more information, see the section "Interest Fee Rate" of the document "Interest Accrual".
- *Interest Fee Account* – account to which fees will be transferred.
- *InterestFee Type* – drop-down list of fee types charged according to the miscellaneous service set up in the contract's Service Package (see the section "Miscellaneous Services" of the document "Way4 Service Packages"). The client is charged the fee after the account interest accrual procedure has been run.
- *Interest Delay* – determines the date starting with which interest is accrued and the date up to which interest is accrued. For more information, see the section "Interest Delay" of the document "Interest Accrual".
- *Calc Int Mode* – allows loan interest to be accrued on the day when the loan is repaid. For more information, see the section "Calc Int Mode" of the document "Interest Accrual".
- *Days In Year* – used to calculate a daily interest rate using an annual interest rate. For more information, see the section "Number of Days in a Year" of the document "Interest Accrual".
- *Interest Contract* – bank contract whose accounts are used during interest accrual. For more information, see the section "Setup of Parameters in Account Templates" of the document "Interest Accrual".

- *Interest Accrual Account, Interest Exp/Rev Account* – bank accounts used during interest accrual. For more details, see the document "Interest Accrual".
- *Supplementary Credit Acc*, *Supplementary Debit Acc* – account pair (debit/credit); when an account of this template is credited, an entry is automatically generated for the same amount between the accounts specified in the *Supplementary Credit Acc* and *Supplementary Debit Acc* fields.
  Usually, these parallel entries are used to repay loan interest. For more information on interest accrual, see the section "Setup of Parameters in Account Templates" of the document "Interest Accrual".
- *Interest Tariff* – drop-down list of tariffs with the "Interest" role registered in the system. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module.

  > The tariff management module is not included in the basic configuration of WAY4 and is provided according to an additional agreement with OpenWay.

3.3.5 GL Properties

This group of fields of the "Full Info for <name of account template>" form (see the figure in the section "Full Information about Account Scheme Templates") is used to determine how an account interacts with the General Ledger.

For more information, see the document "WAY4™ Accounting".

- *GL#* – WAY4 GL account number:
  - The account number can be selected from a drop-down list in the *GL#* field. The list of GL accounts is generated using the "GL Account Plan" dictionary" (Full → General Ledger → GL Accounts → GL Account Plan). The list of GL accounts is filtered by the value of the account template's *GL Type* field, by financial institution (*Fin Institution* field of the account template) and by account currency (*Currency* field of the account template).
    If the field in the "GL Account Plan" form is not filled in, the account number can be selected for any value of the *GL Type* field in the account template (in this case, account numbers are filtered by financial institution and account template currency).
  - The account number can be manually entered in the *GL#* field.

  > The *GL#* field in the template can only be edited if the *GL Type* field is filled in in the account template.

  When the same GL Numbers are used for the account templates of different financial institutions and different currencies, if different names for these accounts are used at the same time, GL account names may be shown incorrectly in the interface. This is due to specifics of data storage and does not influence the correctness of posting transactions to GL accounts.

> When an Account Scheme's *Parent Scheme* field is filled in, the *GL#* and *HeadOffice GL#* fields will not be copied from the parent Account Scheme to the child Scheme if these fields are already filled in in the child Account Scheme. Moreover:
>
> - If the *GL#* and *HeadOffice GL#* fields in the child Account Scheme's template are not filled in, values are inherited from the parent Account Scheme the first time the Scheme is approved.
> - When the *GL#* and *HeadOffice GL#* fields in the child Account Scheme are changed and the Scheme is approved, these values are kept (i.e. not reinherited from the parent Scheme).
> - When the *GL#* and *HeadOffice GL#* fields in the parent Account Scheme are changed, the new values are not inherited to the child Account Scheme.
> - If the child Account Scheme's *GL#* and *HeadOffice GL#* fields are not filled in, on repeat [Approve], values are inherited from the parent Account Scheme.

- *GL Type* – this field defines the accounting area to which the account belongs. An accounting area may correspond to an area of the bank's GL accounting (balance/off-balance) or to an additional technical accounting area (which does not correspond to any area of GL accounting). Possible values:
  - "Balance" – specifies to which balance subsystem of accounting the account belongs to.
  - "Off Balance" – specifies to which off-balance subsystem of accounting the account belongs to.
  - "Technical" – technical accounts used to support Product logic that is not shown in accounting. Used, for example, to create a line of minimum payments.
  - "High precision account" – amounts can be recorded with a high level of precision (without rounding) in accounts with this indicator. For example, to record custom fees in merchant contracts. These accounts belong to the area of additional technical accounting (do not correspond to any GL accounting area).

  When approving an Account Scheme, a check is made that the value of the *GL Type* parameter in the account template matches the value of the *GL Type* parameter set for the corresponding GL account number in the "GL Account Plan" form. If the *GL Type* field in the "GL Account Plan" form is not filled in, a check is not made. The *GL#* field is filtered by the value of the *GL Type* field for new Account Schemes, when adding a new account template. For existing Account Schemes where the template's *GL#* field is already filled in, it is not necessary to fill in the *GL Type* field.
- *Numeration Type* – rules for creating contract accounts and assigning them subsidiary GL account numbers:
  - "First Approval" – a subsidiary GL account number is generated when the corresponding contract is first approved according to a custom procedure.
  - "First Approval" must be used for accounts that directly participate in a transaction's processing (that is, for accounts that are specified in Service parameters; for example,

Merchant Receivable, Merchant Fees). This makes it possible to avoid blocking of contracts when processing operations in parallel.

- "First Transfer" – a subsidiary GL account number is generated when the first posting is made to it according to a custom procedure.
  In a "Main/Sub" hierarchy, if the value is "First Approval", accounts are only created and numbered for the main contract in the hierarchy. This setting does not affect the procedure for creating accounts for subordinate card contracts (for a subordinate card contract, accounts are always created according to the "First Transfer" rule).
- "From GL" – a subsidiary GL account number is set that is equal to the value of the *GL#* field when the corresponding contract is first approved.

It is not recommended to set the "First Transfer" value for accounts that are frequently used (since this may lead to system malfunctions). Examples of frequently used accounts:

- Cl Deposit account
- Accounts specified in Services
- Accounts used in frequently activated normalizations

Setup of a custom account numeration procedure is not included in the standard system setup and is supplied according to a separate agreement with OpenWay.

The way contract accounts are created and subsidiary GL account numbers are assigned to them depends on the following parameters:

- On the value of the global parameter CREATE_ALL_ACCOUNTS (see the document "Way4 Global Parameters"):
  - When this parameter is set to "Y" (Yes), all a contract's accounts are automatically created during contract approval.
  - "N" (No) – during contract approval only those contract's accounts are created whose templates have the value "First Approval" in the *Numeration Type* field. All other accounts are automatically created and numbered as necessary; for example, when a macrotransaction is created that specifies the corresponding contract account. This is the default value of the parameter.
- Accounts for bank contracts are always created during contract approval regardless of the *Numeration Type* field value in an account template. When account templates are created for bank contracts, the "First Approval" value is automatically set in the *Numeration Type* field for information.
  The procedure for creating and numbering contract's accounts can be additionally configured on the account template level using the CREATE tag in the *Template Details* field (this setting is possible when the value of the global parameter CREATE_ALL_ACCOUNTS is "N"):
  - When the "MAIN" value is set, main contract accounts are created when the corresponding contract is first approved (or when the Account Scheme is approved, if it is approved after contract approval).
  - When the value is "ALL", accounts for both the main contract and subcontracts are created on the first Approval of the corresponding contract (or when an Account Scheme is approved if it is approved after a contract).

- *Aggregate GL For* (until version 03.35.30 the *Use GL#* field):
  - "Sub GL" (until version 03.35.30, "For Analytic") – when a contract account is created, the number of a subsidiary GL account is used as the GL account number. In this way, for each contract account (or group of contract accounts with one subsidiary GL account number), a separate GL account is generated. "Sub GL" can be used, for example, for bank contracts when several contracts have the same Account Scheme but when generating GL entries, for each contract its own number must be used. In this case, "Sub GL" is set for the account template, and the account is given the required number in the "Renumber Subsidiary GL Accounts" form (Full → DB Administrator Utilities → Special Contract Utilities → Renumber Subsidiary GL Accounts). When a subsidiary GL number is set, the same number will be specified in the account's *GL Number* field.
  - "Consolidated GL" (until version 03.35.30 – "For Synthetic") – the GL account number will be taken from the *GL#* field; that is, contract accounts created according to this template will correspond to one GL account.

    - Empty (null) – the same as "Consolidated GL".
- *Off-Balance XF Acc* – drop-down list of accounts of the contract specified in the *Interest Contract* field of the account template. Used when it is necessary to transfer funds from balance accounting to off-balance accounting (see "Off-Balance Accounting Subsystem").
- *FX Type* – conversion rate type. For more details on currency conversion in Way4, see the document "Currency Conversion".
- *GL Tariff* – field with a drop-down list of registered tariff types with the "GL Numeration" role.

> The Advanced Tariff Management module can be used to change a GL account number. The Advanced Tariff Management module is not included in the WAY4 basic configuration and is delivered according to an additional agreement with OpenWay.

- *Reference Mode* – used to configure synchronization of the main Scheme's templates with the templates of an additional Account Scheme. For more information, see the section "Working with Included Account Schemes".
- *Template Details* – this field is used to specify additional parameters of the account template as tags. See the section "Tags used when configuring Account Schemes and account templates" of the "Setup Tags" document.

3.4 Off-Balance Accounting Subsystem

Data in both balance and off-balance subsystems is closed, that is, accounting entries cannot be performed between accounts in different subsystems. Therefore, when it is necessary to transfer funds from balance accounting to off-balance accounting, a sequence of two actions is performed: funds are returned to their source account in the balance accounting subsystem and a balance is created by debiting (or crediting) the corresponding off-balance account in correspondence with special accounts (for example, 99999 and 99998 in Russia). The opposite operation (returning funds from off-balance accounting to balance accounting) is performed in the same way.

For this, the following parameters can be specified for each account template in an Account Scheme in WAY4:

- Accounting subsystem (balance, off-balance) to which the account belongs. The *GL Type* field of an account template is used for this. This field can have one of the following values: "Balance" – the account belongs to the balance subsystem; "Off-Balance" , "High precision account" – the account belongs to the off-balance subsystem.
- Code of the account with which the current account will correspond when funds are transferred to the other financial subsystem. This parameter can be described in one of the following ways:
    - In the *OffBalance XF Acc* field, an account is selected from the list of accounts of the contract specified in the *Interest Contract* field of the account template. This field of a balance contract account is used to specify the corresponding balance account of a bank contract (specified in the *Interest Contract* field of the account template); for a template of

an off-balance account, this field contains the corresponding off-balance account of a bank contract.

- The *Template Details* field contains a tag for determining a pair of accounts that participate in entries during interaction between balance and off-balance accounting subsystems. The *Template Details* field is filled in according to the following rules. Field format: "XF=<Off_1>:<Bal_1>, ...,<Off_N>:<Bal_M>,ELSE:<Bal_Default>;"The following notation is used in this record: Bal_N – bank contract balance account type code. Off_N – card contract off-balance account type code. <Bal_Default> – default bank contract balance account type code.

Only one of the above fields may be filled in in an account template. The *OffBalance XF Acc* field is filled in if the same corresponding balance account of a bank contract is used regardless of the off-balance account to which funds must be transferred. If the balance account of a bank account depends on the off-balance account to which funds must be transferred (for example, when different income accounts must be used), the *Template Details* field is filled in.

If the *OffBalance XF Acc* and *Template Details* fields are empty, the BALANCE_XF_CODE and OFF_BALANCE_XF_CODE global parameters are used to determine the corresponding accounts.

Example 1:

When a contract goes from behavior type 1 to behavior type 2 or higher, loan interest accrued to the interest account must be transferred to the off-balance account of unearned interest.

A macrotransaction is generated in the system, containing a link to the "Cl Loan Int" account of the card contract in the *Source Account* field and a link to the "Cl Loan Int Off Bal" account of the card contract in the *Target Account* field.

Descriptions of the accounts show that "Cl Loan Int" belongs to the balance subsystem, and "Cl Loan Int Off Bal" belongs to the off-balance subsystem. If the *OffBalance XF Acc* field of the "Cl Loan Int" account template contains a link to the corresponding account of a bank contract, the following entries shown in the "GL_TRACE" system table are made:

- Debiting the "Cl Loan Int" account of the card contract → crediting the corresponding balance account. This account belongs to the interest contract specified in the *Interest Contract* field of the "Cl Loan Int" account template. The name of the account is specified in the *OffBalance XF Acc* field of the "Cl Loan Int" account template.
- Debiting the off-balance correspondent account → crediting the "Cl Loan Int Off Bal" account. The correspondent account belongs to the interest contract specified in the *Interest Contract* field of the card contract's off-balance account template. The name of the account is specified in the *OffBalance XF Acc* field of the "Cl Loan Int Off Bal" account template.

Example 2.

When a contract goes from behavior type 1 to behavior type 2, loan interest accrued to the "Cl Loan Int" account must be transferred to the off-balance account "Cl Loan Inf Off 2". When a contract goes from behavior type 1 to behavior type 3 or 4, loan interest accrued to the "Cl Loan Int" account must be

transferred to a separate off-balance account of unearned interest, for example "Cl Loan Inf Off 3-4". Different balance accounts of a bank contract must be used in different cases.

For example, account types shown in table are registered in the system.

List of registered accounts showing interaction with the off-balance accounting subsystem

Product Category	Name	Code
Bank Accounting	Revenue 2	k
Bank Accounting	Revenue 3-4	55
Bank Accounting	Revenue Other	m
Issuing	Cl Loan Inf Off 2	8
Issuing	Cl Loan Inf Off 3-4	j

In this case, the *OffBalance XF Acc* field of the "Cl Loan Int" account template must not be filled in; account data for generating entries is taken from the *Template Details* field of the "Cl Loan Int" account template.

Assume the field contains the "XF=8:k,j:55,ELSE:m;" tag.

When a contract goes from behavior type 1 to behavior type 2, the following entries will be generated as a result of processing of a standing payment order or a financial document transferring funds from the "Cl Loan Int" balance account to the "Cl Loan Inf Off 2" off-balance account:

- Debiting the card contract's account "Cl Loan Int" → crediting the "Revenue 2" account. This account belongs to the interest contract specified in the Interest Contract field of the "Cl Loan Int" account template. The account's type code ("k") is specified in the "8:k" account pair, this pair is selected because the "Cl Loan Inf Off 2" account type has code "8".
- Debiting the off-balance correspondent account → crediting the card contract's off-balance account. The correspondent account belongs to the interest contract specified in the Interest Contract field. The name of the account is specified in the OffBalance XF Acc field of the "Cl Loan Inf Off 2" account.

When a contract goes from behavior type 1 to behavior type 3 or 4, the following entries will be generated as a result of processing of a standing payment order or a financial document transferring funds from the "Cl Loan Int" balance account to the "Cl Loan Inf Off 3-4" off-balance account:

- Debiting the card contract's account "Cl Loan Int" → crediting the "Revenue 34" account. This account belongs to the interest contract specified in the *Interest Contract* field of the "Cl Loan Int" account template. The account's type code ("55") is specified in the "j:55" account pair, this pair is selected because the "Cl Loan Inf Off 3-4" account type has code "j".
- Debiting the off-balance correspondent account → crediting the card contract's off-balance account. The correspondent account belongs to the interest contract specified in the *Interest*

Contract field. The name of the account is specified in the *OffBalance XF Acc* field of the "Cl Loan Inf Off -4" account.

When funds are transferred from the "Cl Loan Int" balance account to an off-balance account whose code is not used in any code pair, the default account will be used. In this example, its code is "m".

3.5 Configuring Display of Past Due Date and Past Due Days in Customer Service Workbench

There are two ways to configure calculation of data to be shown in the *Past Due Date* and *Past Due Days* fields in Customer Service Workbench(Customer Service → Customer Service):

- **Display of fixed data about the date delinquency arose and the number of past due days until the time delinquency is fully paid.** The date delinquency arose is set in the *Past Due Date* field when funds are transferred from a standard account to a delinquency account (when the balance specified in the PAST_DUE_BALANCE tag is opened, see the settings below). **This date is not corrected if the debt is partially repaid**. That is, the date funds are first transferred from the standard account to the delinquency account is shown in the *Past Due Date* field until the debt is fully repaid. The total number of days in the *Past Due Days* field is calculated from this date (that is, from the date the balance specified in the PAST_DUE_BALANCE tag was opened, see the settings below). Display of the past due date in this way is configured as follows:
- Configure a separate balance type to record the total amount of debt. To do so, it is recommended to use the PAST_DUE balance type (see figure).

Balance Types — 9 of 9

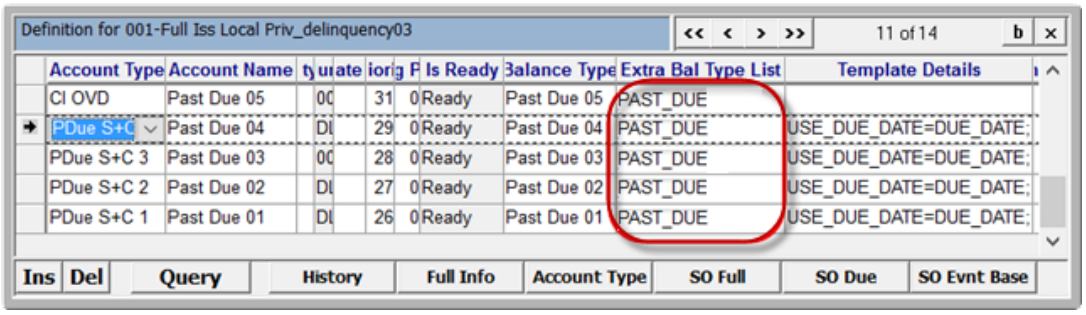
Name	Code	Main Only	History Mode	Is State	Dlq Code	Dlq Level	Skip Liab	Direction	Group Code	Posting Details
Past Due 08	OVD_08	Main Only	None		8	9		None		USAGE=BALANCE;
Past Due 07	OVD_07	Main Only	None		7	8		None		USAGE=BALANCE;
Past Due 06	OVD_06	Main Only	None		6	7		None		USAGE=BALANCE;
Past Due 05	OVD_05	Main Only	None		5	6		None		USAGE=BALANCE;
Past Due 04	OVD_04	Main Only	None		4	5		None		USAGE=BALANCE;
Past Due 03	OVD_03	Main Only	None		3	4		None		USAGE=BALANCE;
Past Due 02	OVD_02	Main Only	None		2	3		None		USAGE=BALANCE;
Past Due 01	OVD_01	Main Only	None	Open/Close	1	2		None		USAGE=BALANCE;
All past due	PAST_DUE	Main Only	None			0		None		USAGE=BALANCE;

Ins | Del | Query | Do ... | SubBal of | Includes

Configuring a balance type to record the total amount of debt

- Map all delinquency account templates with this balance type, see figure.

> Note that these account templates are also mapped with balance types for recording delinquency when configuring the standard DLQ_LEVEL classifier, see the section "Configuring the "DLQ_LEVEL" System Classifier" of the document "WAY4™ Client and Contract Classifiers".



Mapping account templates with the PAST_DUE balance type

- This balance type's code should be specified as the value of the PAST_DUE_BALANCE global parameter.
- **Changing the date that is displayed for when delinquency arose, in the case of partial early repayment**. The date on which delinquency arises is set in the *Past Due Date* field when funds are transferred from a standard account to a delinquency account (this date is determined according to a contract functional date with the code PD_DATE, see the settings below). If the debt is partially paid, the date is corrected: this functionality uses invoices (special technical records in the invoice_log table) and the *Past Due Date* field will show the date of the oldest unpaid invoice related to transfer of funds to a delinquency account. The total number of past due dates (in the *Past Due Days* field) that is calculated from the date of partial payment of the debt is also corrected.
  For example, delinquency arose on 01 March. This date is shown in the *Past Due Date field.* On 15 March, the total number of past due days shown in the *Past Due Days* field is 15 days. On 16 March, the debt is partially paid. A date in the *Past Due Date* field changes to 16 March, and a new value is also set in the *Past Due Days* field – 1 (day).
  Display of the past due date in this way is configured as follows:
- Specify the CALC_PD_DATE tag in the Account Scheme's *Special Parms* field.
- Delinquency accounts must be mapped with balance types when configuring the DLQ_LEVEL classifier, see the section "Configuring the "DLQ_LEVEL" System Classifier" of the document "WAY4™ Client and Contract Classifiers".
- A custom contract functional date with the PD_DATE code is registered by default in WAY4. This functional date is used to show the date and term of delinquency for accounts mapped by with the DLQ_LEVEL classifier in the *Past Due Date* and *Past Due Days* fields in Customer Service Workbench (Customer Service → Customer Service). For more information about functional dates, see the document "Contract Functional Dates".

  Calculation of the date and number of past due days using invoices (with the ability to correct the date shown for which delinquency arose when partial payment is made) is only possible if the Reversal Management module is used. The Reversal Management module is required to correctly process reversals of deposits that repay a debt.

This functionality may be used starting from version 03.44.30. Note that after the aforementioned settings have been made, the new approach is applied to new delinquency. Data for old delinquency (that arose before these settings were made) do not change (are not recalculated).

4 Working with Account Schemes

Working with Account Schemes consists of the following operations:

- Copying Account Schemes
- Configuring Message Templates (Group Msg)
- Configuring Events
- Entering and Editing Tags
- Checking Account Schemes
- Account Scheme Approval

4.1 Copying Account Schemes

An Account Scheme can be created in the following ways:

- By clicking the [Ins] button to add necessary information.
- By copying an existing Scheme and changing its parameter values according to the requirements to a new Account Scheme:
- Clicking the [Details] button in the "Account Schemes" grid form opens the form containing additional information about the Scheme.
- In the additional information form, click the [Actions...] button and select the "Duplicate" value from the context menu; after duplication, a new Account Scheme is added to the list; its name is the name of the copied Scheme with the underline character as a prefix.
- Change the name of the Scheme and other necessary parameters.
- After modifying Scheme parameters, approve the Scheme (see the section "Account Scheme Approval").

4.2 Configuring Message Templates (Group Msg)

In WAY4, it is possible to send information or marketing messages to clients whose contracts use a specific Account Scheme.

To set up a template of this message, use the [Group Msg] button in the "Details for <name of Account Scheme>" form (see the figure in the section "Additional Account Scheme Parameters"). For more information about configuring message templates, see the document "Configuration of Client Messages".

4.3 Configuring Events

Events for Account Schemes are configured in the "Events for <name of Account Scheme>" form, opened by clicking the [Events] button in the "Account Schemes" form (see the figure in the section ""Account Schemes" Form").

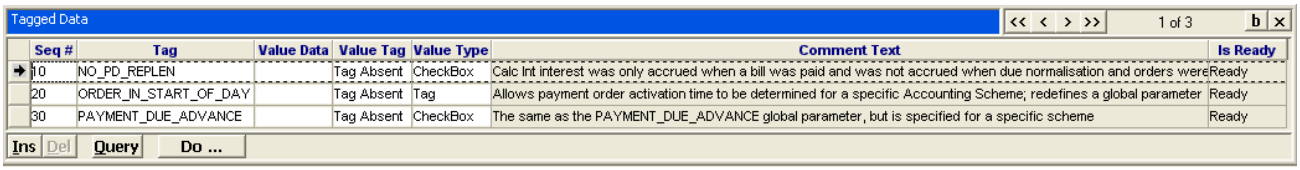
This form is used to support the following functions:

- The form is used to configure Events for changing a contract's Behavior Type. For more information about setting up the Event, see the section "Changing Contract Behavior Types" of the document "Events".
- An Event to change the interest rate on the contract account (when the interest rate remains unchanged in the account template) must be registered in this form. For more information, see the section "Event Types" of the document "Events").
- This form is used when configuring an Event intended to change the Account Scheme. For more information see the section "Changing a Contract Account Scheme" of the document "Events".

4.4 Entering and Editing Tags

Forms "Details for <name of Account Scheme>" and "Full Info for <name of account template>" contain special fields for entering and editing tags – *Special Parms* and *Template Details*, respectively.

The "Tagged Data" form (see figure) is used to optimise the process of entering and editing tags in the above fields. The form is opened by clicking the [Tagged Data] button in forms "Details for <name of Account Scheme>" and "Full Info for <name of account template>".


Tagged Data

<< < > >> 1 of 3

Seq #	Tag	Value Data	Value Tag	Value Type	Comment Text	Is Ready
10	NO_PD_REPLEN		Tag Absent	CheckBox	Calc Int interest was only accrued when a bill was paid and was not accrued when due normalisation and orders were	Ready
20	ORDER_IN_START_OF_DAY		Tag Absent	Tag	Allows payment order activation time to be determined for a specific Accounting Scheme; redefines a global parameter	Ready
30	PAYMENT_DUE_ADVANCE		Tag Absent	CheckBox	The same as the PAYMENT_DUE_ADVANCE global parameter, but is specified for a specific scheme	Ready

Ins Del Query Do ...

"Tagged Data" form

To add a tag, click the [Ins] button and fill in the fields of a new record:

- A tag name in the *Tag* field can be selected from the system list of tags. If a tag is absent from the list, its name can be entered from the keyboard.
- In the *Value Data* field, enter a tag value.
- The *Value Tag* field:
- When entering tag parameters in the "Tagged Data" form, select the "Tag Present" value in the *Value Tag* field so that tag parameters are saved correctly. After data is saved, tag parameters will be displayed in the corresponding field of the parent form (in field *Special Parms* or *Template Details*).
- When the "Tag Absent" value is selected, the tag record will be deleted from the "Tagged Data" form as well as from the corresponding field of the parent form after changes are saved.
- The *Value Type* field is used to determine the type of the tag value entered in the *Value Data* field:
- "CheckBox" – in this case, it is not necessary to fill in the *Value Data* field
- "String" – string value.
- "Counter" – integer value (from "0" to "9").

- "Tag" – the value of the tag must be either "Y" or "N".
- "Money" – numeric value.
- "Currency" – the value of the tag must be a numeric currency code.
- "Unknown" – no tag value type is specified, but the *Value Data* field must be filled in.
- "List" – the tag value may be set as a list of values. In initial manual setup, a comma-delimited list of values is specified in the *Value Data* field (or in the *Special Parms* field of the higher-ranking Account Scheme form). After data is saved, each "List" type tag value will be displayed as a separate record in the "Tagged Data" form.
- The *Comment Text* field contains a description of the tag whose name is selected from the system list.
- The *Is Ready* field shows results of tag parameter check:
- The field contains the "Ready" value if the check is successful.
- The field contains the "Not Ready" value if errors have been detected during the check.

To check tag parameters for correctness, click the [Do...] button and select the "Check" value from the context menu. If an error is detected, a window with the corresponding message will open.

To save entered data, click the [Do...] button and select the "Save Tags" value from the context menu.

> Tag parameters entered in "Tagged Data" can also be saved by clicking the [Actions...] button and selecting from the context menu:
>
> - "Save Tagged Data" in the "Details for <name of Account Scheme>" form.
> - "Save Tags" (in the "Full Info for <name of account template>" form).

4.5 Checking Account Schemes

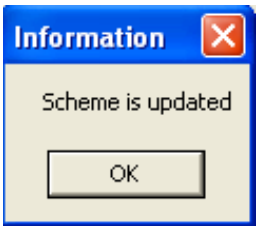
Account Scheme parameters are checked for correctness in the form containing additional Account Scheme information (see the figure in the section "Additional Account Scheme Parameters") by clicking the [Actions...] button and selecting the "Check" value from the context menu.

If errors are detected while checking Account Scheme parameters, a window with the "Error" heading is displayed on the screen. Information about errors is available in the process log opened, for instance, by selecting "Full → Process Log → Last Process" from the user menu. This will open the "Last Process" form with information about execution of the last process in WAY4.

4.6 Account Scheme Approval

After entering or modifying values of Account Scheme parameters, click the [Approve] button in the Account Scheme grid form (see the figure in the section ""Account Schemes" Form" or the figure in the section "Special Forms for Working with Account Schemes") to approve the changes.

If the entered information is correct, the following message will be displayed on the screen:



Message informing that Account Scheme data has been updated successfully

When an Account Scheme is modified for which contracts have already registered and accounts have already opened in the system, the properties of the contracts and accounts will be updated according to the value of the APPROVE_IMMEDIATE global parameter – immediately, during approval, or when a special menu item is run (for more details, see the document "Way4 Global Parameters").

While changes to an Account Scheme are approved, the parameters of the Scheme are automatically checked. If errors are detected while checking Account Scheme parameters, a corresponding message is displayed on the screen (for more details, see the section "Checking Account Schemes").

The global parameter CHECK_ACC_SCHEME_CODE makes it possible to check the uniqueness of an Account Scheme code when approving the Account Scheme.

If in an Account Scheme an account template is deleted according to which accounts have already been created (records in the ACCOUNT table), and activity has taken place in these accounts, this Account Scheme cannot be approved. An error message will be displayed: "Account Template cannot be deleted, you should manually undelete it. Create new Account Scheme by duplication if needed". The deleted account template record must be restored to successfully approve the Account Scheme.

If any parameters related to calculation of billing dates change (specified in Account Schemes, date schemes, tariffs, etc.), the Billing Date will be recalculated when the Account Scheme is approved.

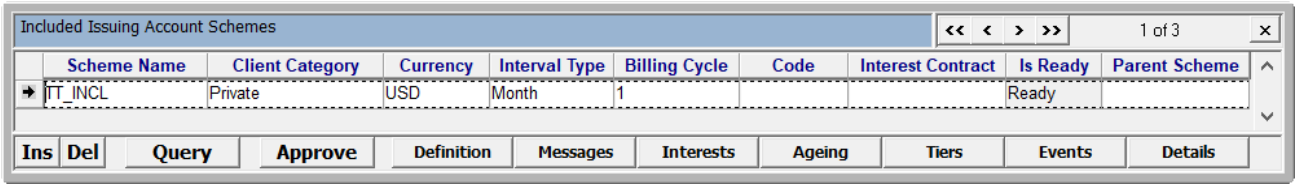
Any changes in rules for calculating billing dates must be approved by OpenWay and must have been tested in advance.

4.7 Working with Included Account Schemes

WAY4 provides functionality for attaching sets of additional accounts to main Account Schemes. To do so, do as follows:

- Configure an additional Account Scheme with a set of required accounts. The following menu items are used to do so:

- "Full → Configuration Setup → Products → Issuing Private Products → Included Issuing Account Schemes" – Account Schemes for the issuing contracts of private clients.
- "Full → Configuration Setup → Products → Issuing Corporate Products → Included Issuing Account Schemes" – Account Schemes for corporate issuing contracts.



Form for registering an included Account Scheme

- Form fields are described in the section ""Account Schemes" Form". It is not necessary to specify client category (*Client Category* field) for an included Account Scheme. These schemes can also be included regardless of the main Scheme's client category. When adding an included Account Scheme using the aforementioned menu items, the *Used For* field is automatically filled in with "Included". The *Used For* field is not shown in the "Included Issuing Account Schemes" form. The *Used For* field with the "Included" values is shown for included Schemes in the form "Full → Configuration Setup → Products → Account Schemes" (see the section ""Account Schemes" Form").
- Attach the configured Account Scheme to the main Scheme. To do so, do as follows:
- Open the form with additional parameters of the main Account Scheme to which the configured set of accounts should be attached (the value in the *Used For* field of this Account Scheme is either empty or the "Product" value is specified). See the section "Additional Account Scheme Parameters".
- In the form with additional parameters of the main Account Scheme, click the [Included] button.
- In the "Included for <name of main Account Scheme>" form that opens, use the [Ins] button to add a new record and in the *Included Acc Scheme* field, select the name of the included Account Scheme.

> Only Schemes with the "Included" marker may be included in Account Schemes.

When approving a main Account Scheme for which an included Scheme is set, the account templates of this Scheme are synchronised with the attached Scheme:

- Missing account templates are copied from the attached Scheme to the main Account Scheme.
- If in the main Account Scheme there is an account template with the same type as in the included Account Scheme, the settings of the main Scheme template will be changed according to the parameters of the attached Scheme template.

A number of the main Scheme's template parameters can be redefined. The *Reference Mode* field of the main Scheme's templates is used to set up this functionality. Possible values for the field:

- "Reference Only" – if the field is not filled in or is filled in with the "Reference Only" value, when the parameters of the template for a certain type of account change in the main Account Scheme, this template's settings (that is, the template of the main Scheme) will be changed accordingly.
- "Redefinition" – when this value is specified, the following parameters can be redefined in the template of the main Scheme: *Balance Type*, *Extra Bal Type List*, *Payment Priority*, *Ageing Priority* fields, INT_GROUP tag value. That is, the values of these parameters will not be synchronized with the attached template's parameters.

When the INT_GROUP tag is specified in the main Account Scheme's template, a check is made in the main Scheme for the templates of all accounts listed in this tag.

- When changing account template parameters in a Scheme that is already included, these changes are automatically inherited by the main Account Scheme when it is approved.
  Included Schemes are deleted using the "Included for <name of main Account Scheme>" form. When deleting an included Account Scheme, account templates of the main Scheme that were added or changed according to the templates of an additional Scheme are not deleted from the main Scheme. When deleting an included Scheme, the link between the main Scheme and attached Scheme templates is deleted. Synchronization of template changes is terminated and access to changing parameters of templates that were included earlier is opened right in the main Account Scheme.

4.8 Template Included Account Schemes

A template included Account Scheme is a scheme with a template currency that is replaced with the main Scheme's currency when attaching to the main Account Scheme. See the section "Working with Included Account Schemes".

Template Schemes are used to optimize the process for creating Account Schemes and making changes to them. A template Scheme as the standard included Account Scheme is used to attach additional accounts to the main Account Scheme. One template Scheme makes it possible to attach a set of accounts of the same type to several main Account Schemes in different currencies. A template Scheme contains account templates with general rules for recording operations that are effective for main Account Schemes. Parameters for certain Account Schemes, such as fee rates or GL account numbers are configured using tariffs. If it is necessary to make changes to parameters of account templates, make changes to one template Account Scheme.

For an included Account Scheme, specify the TEMPL_CURR tag in the *Special Parms* field and set a template currency in the *Currency* field.

> A template Scheme must contain accounts in the template currency only. If template accounts contain several currencies, only one currency that corresponds to a template currency will be replaced when attaching to the main Account Scheme.

Template Schemes can be used when configuring Account Schemes of the same type in different currencies (see the section "Working with Account Schemes of the same type in different currencies").

4.9 Working with Account Schemes of the same type in different currencies

The setup is used if contracts in different currencies can be created at different levels of a contract hierarchy. This makes it possible to avoid creating Products for each currency.

For example, it is necessary to set up Products for a three-level contract hierarchy. Five currencies can be used at each level. To do so, three Products are configured using this setup. Five Account Schemes in different currencies are created for each Product. Without this setup, 125 Products have to be created. That is, 5 parent Products at the top level (in each currency), 5 child Products (in each currency) and 5 Products for each child Products.

Setup procedure:

- Create a Product with the SCHEME_BY_CUR tag in the *Apply Rules* field. The tag contains a list of currencies for which this Product can be used. For example, SCHEME_BY_CURR=USD,RUR,EUR;.
- Create Account Schemes:
  - The number of Account Schemes is equal to the number of currencies that are supposed to be used when creating contracts.
  - An Account Scheme code is specified in <code>_<currency name> format. Codes of all Schemes for this Product must differ by a postfix with the currency name. For example, SCHEME_CODE_USD, SCHEME_CODE_EUR.
  - Account Schemes have the same type and can be created as follows:
    - Copying an Account Scheme changing the currency (*Currency* field) and Scheme code (*Scheme Code* field).
    - Using template included Account Schemes (see the section "Working with Included Account Schemes"). In this case, main Account Schemes do not contain account template settings. Account templates are attached using a template Account Scheme and inherit the main Scheme's currency.
- A contract with this Product is created when receiving an application. The application must contain the contract currency. When a contract is created, an Account Scheme with the specified currency is selected from a list of Account Schemes. That is, the Account Scheme is defined when a contract is created.
- Tariffs (for example, a tariff for specifying GL account numbers) can be attached depending on the selected currency. To do so, use a subdomain hierarchy with the

ATTACH_BY_CODE=CONTRACT.CURR parameter or check IF_PARM=CONTRACT.CURR. Registering a custom parameter and tariff plan, and setting the default parameter value for the Product is not required.

5 High Precision Accounting

"High Precision" accounts are WAY4 technical internal accounts that do not participate in standard accounting. These accounts are used to record cash flows with a high degree of precision (up to 10 decimal places).

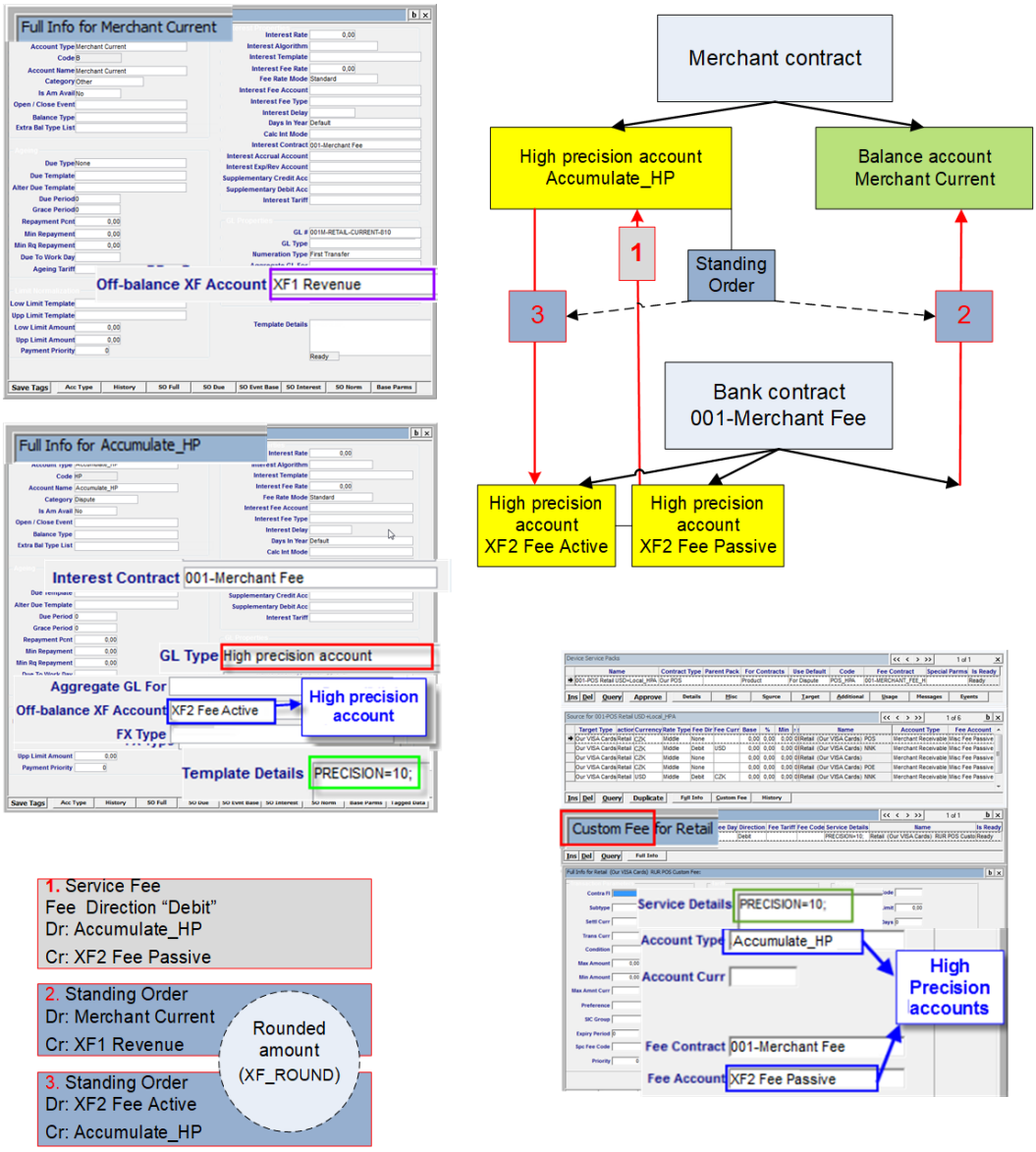
"High Precision" accounts are used to accumulate amounts with high precision and then show them in regular balance accounts. They can be used to record custom fees with high precision.

Custom fees are accumulated in a high precision account and then a payment order is used to make a consolidated entry to a standard balance account (the entry is made indirectly, through an XF account).

When an amount is "transferred" to a standard balance account, the amount is rounded to two decimal places. The difference between the original and rounded amount may remain in the "High Precision" account or can be fully debited (depending on settings).

For an example of recording issuer high precision interchange fee, see the section "Issuing interchange fee" of the document "Payment system settlement in Way4™".

5.1 General Scheme of Operation



Example of general account setup scheme for high precision recording of fees (without settings in the GL_TYPE table)

5.2 Service Settings

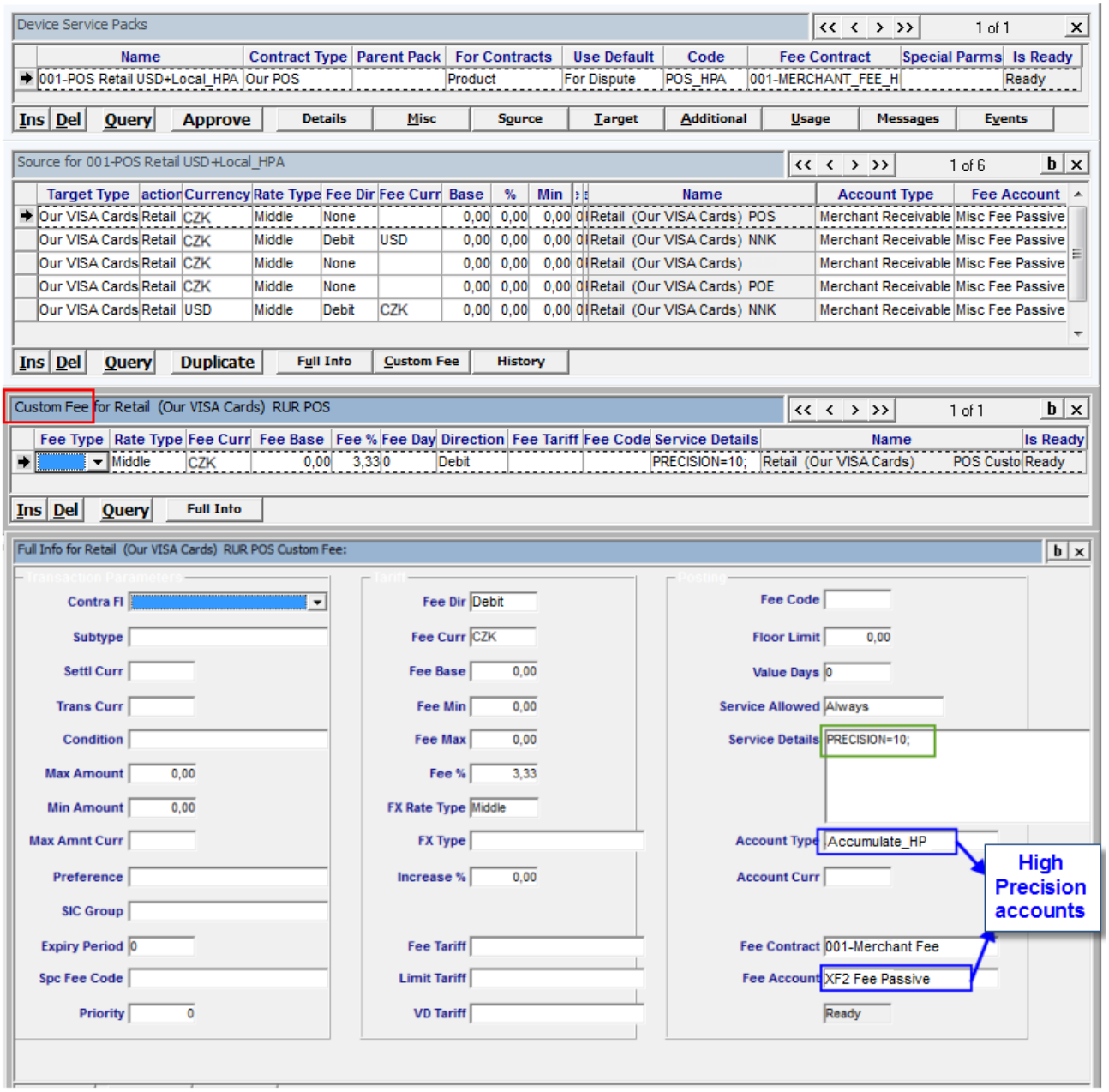
Additional settings for the Service of a high precision custom fee (see figure):

- The PRECISION=<number of decimal places>; tag is set in the custom fee's *Service Details* field. This tag determines the number of decimal places for the fee amount that will be saved (see figure).
- If a custom fee's *Service Details* field has the PRECISION=<N>; tag, and the fee has a high precision account and standard balance account:
    - The entry between a high precision account and XF account is posted with the precision specified in a Service.

  - The entry between the standard account and XF account is posted according to the currency exponent (rounding to fractional currency units).

- The *Account Type* field specifies the merchant contract account from which the custom fee will be debited (cumulative account). An account type with the "High precision" marker should be specified in the *Account Type* field. For information on setting up this account, see the section "Configuring High Precision Account Templates".
- In the *Fee Account* field (bank contract account to which the custom fee will be transferred), specify an account type with the "High precision" marker. For information on setting up this account, see the section "Configuring High Precision Account Templates".

The figure shows an example of setup for charging a custom fee (with high precision) from a merchant contract.



Settings for a high precision custom fee

5.3 Configuring High Precision Account Templates

High precision amounts can only be records in accounts with the "High Precision" marker.

An entry with a high precision amount can only be made if the entry is made between accounts with the "High Precision" marker. Otherwise, standard rounding to fractional currency units is performed (for example, to two decimal points for dollars).

Therefore, to record fees with high precision, the following accounts with the "High Precision" marker must be created:

- The account from which the custom fee is debited (this account type is specified in the custom fee's *Account Type* field, in the figure of the section "Service Settings" this is the "Accumulate HP" account). This account type is searched for in the accounts of the main settlement contract for the transaction. That is,if a fee is debited from a merchant's account, a special account of the defined type with the "High Precision" marker must be created. In this account, a payment order is set up for transferring the accumulated fee amount to a standard account (see the section "Setting up a Payment Order (Transferring an Accumulated Amount to a Standard Account)").
- The bank contract "High Precision" account to which the custom fee is transferred. This account is specified in the custom fee's *Fee Account* field (see the figure in the section "Service Settings"). That is, a special account with the "High Precision" marker must be created for the banking contract from the *Fee Contract* field. This is usually a common account for correspondence with any "High Precision" accounts (in the example in the figure in the section "Service Settings", this is the "XF2 Fee Passive" account of the 001-Merchant Fee bank contract).

  > When setting up bank contract accounts to record fees with high precision, an asset/liability account pair with the "High Precision" marker is set up (see the figure in the section "General Scheme of Operation").

"High Precision" account setup (see figure):

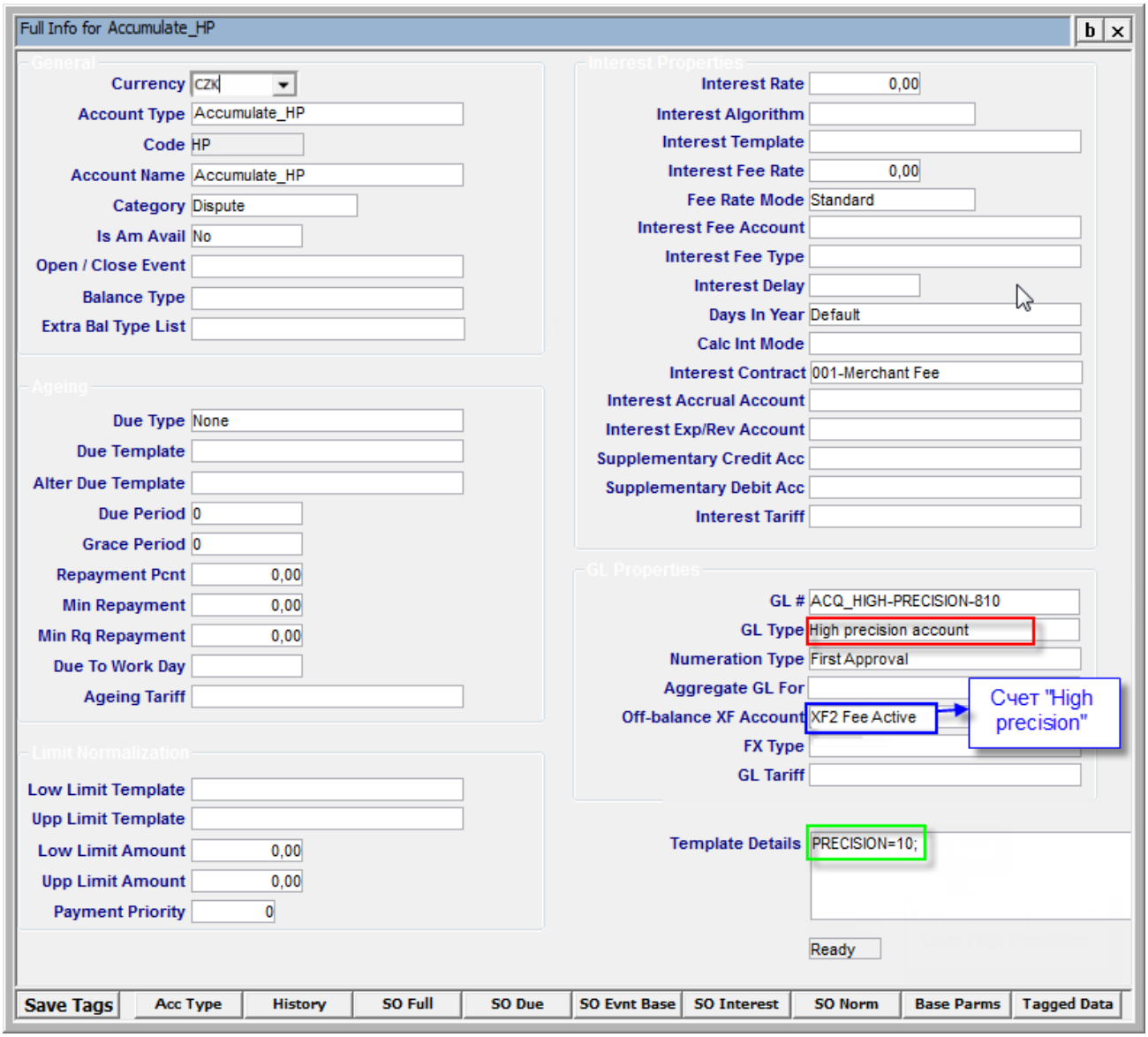
- To specify the "High Precision" marker, select the value "High Precision account" in the account template's *GL Type* field.
- In the GL # field of the account template with the "High Precision" marker, specify a special GL account that is not used in standard accounting.
- "High Precision" accounts cannot participate in calculating a contract's balance. That is, when setting up the corresponding account type, "Yes" may not be specified in the *Is Am Av* field (if "Yes" is specified, the Account Scheme will not be approved).

XF accounts are additionally configured for transfers between a "High Precision" account and a standard merchant contract account:

- An OffBalance XF Account must be set for a "High Precision" account – this is a consolidated account for correspondence with "High Precision" accounts.

- For the standard balance account to which the accumulated fee amount is transferred by the order, an OffBalance XF Account must be set up corresponding to the actual revenue account for this fee.

For an entry in which a balance account and a "High Precision" account participate, the entry is automatically separated into two parts – Balance and High Precision.



Template of a merchant contract's "High Precision" cumulative account for recording a custom fee with high precision

XF accounts are set up in the standard way (see the section "Off-Balance Accounting Subsystem").

In addition to XF accounts explicitly specified in Account Scheme templates, rules for defining an XF account can be set as follows:

- Determining the contract in which an XF account is selected:
- If the OWN_XF_ACCOUNT; tag is specified in the *Template Details* field of a "High Precision" account template, or in the *XF Configuration* field of the "GL Type" form for the "High Precision"

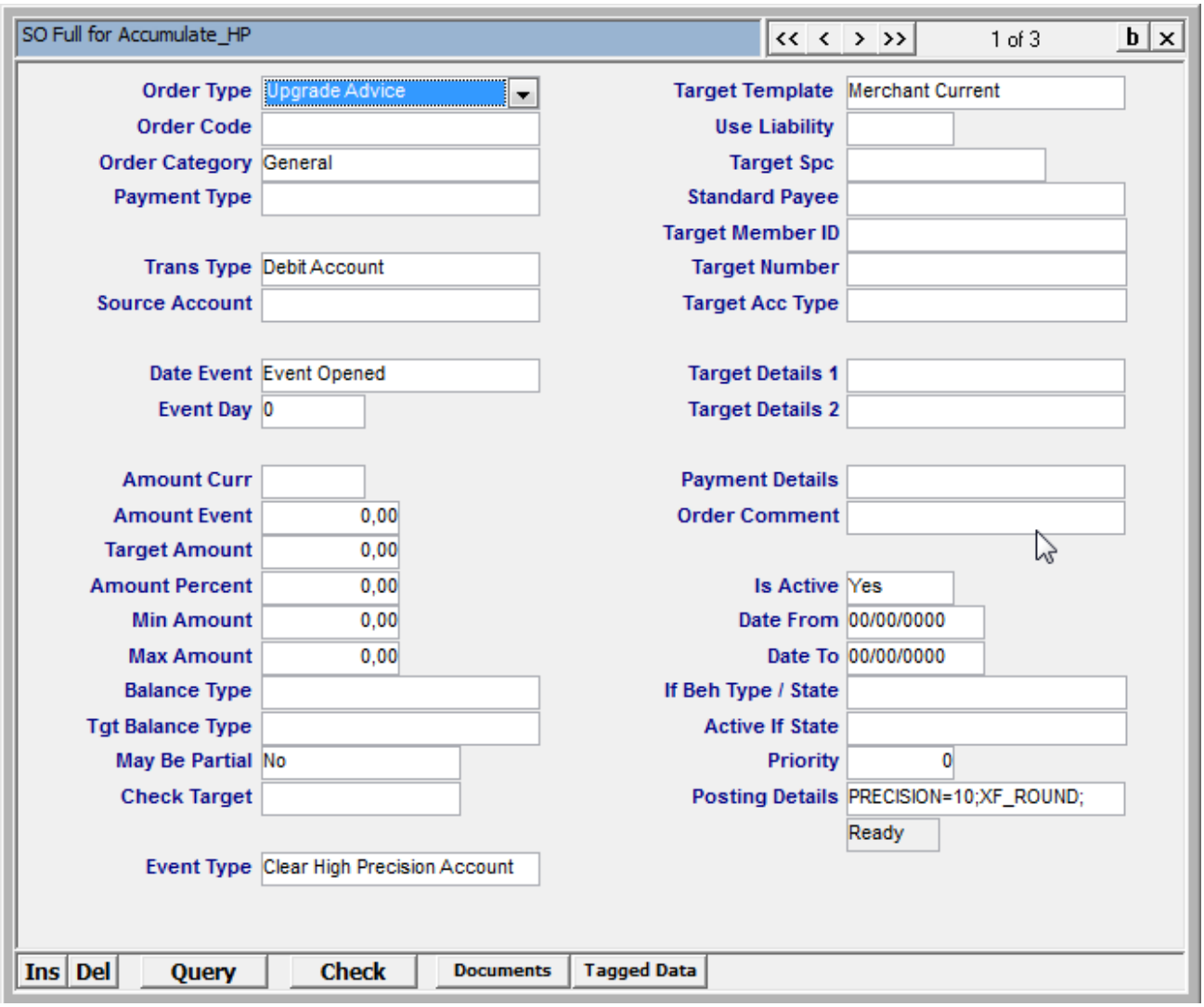
accounting section (see the section "Configuring the "GL Types" Form"), the XF account for this account will be selected in the same contract as the original account.

- If the OWN_ALT_XF_ACCOUNT; tag is specified in the *Template Details* field of a "High Precision" account template, or in the *XF Configuration* field of the "GL Type" form for the "High Precision" accounting section (see the section "Configuring the "GL Types" Form"), the XF account for the counterparty contract (XF account for the balance account) will be selected in this contract and not in the counterparty contract.
- If the ALT_XF_ACCOUNT; tag is specified in the *Template Details* field of a balance account template, or in the *XF Configuration* field of the "GL Type" form for the corresponding accounting section (see the section "Configuring the "GL Types" Form"), the XF account for this balance account will be selected in the same contract as the original (balance) account.
- If the XF_CONTRACT tag is specified in the *XF Configuration* field of the "GL Type" form for the "High Precision" section (see the section "Configuring the "GL Types" Form"), two values are possible: FI_DEPOSIT and FI_DISPUTE. The XF account will be selected from the financial institution's Deposit Contract or Dispute Contract, respectively. This tag is not provided for in the account template.
- If the OWN_XF_ACCOUNT, OWN_ALT_XF_ACCOUNT, ALT_XF_ACCOUNT, and XF_CONTRACT tags are not specified, the interest contract specified in the original account will be used to search for the XF account.
- Determining the account type (in descending order of the priority):
- Determining an XF account based on account template settings – explicit specification of an XF account in the account template's *Off Balance XF Acc* field or using the XF=<>; tag in the account template's *Template Details* field.
- The tag XF_<code of the corresponding accounting section>=<>; in the *XF Configuration* field of the "GL Type" form (see the sections "Off-Balance Accounting Subsystem", "Configuring the "GL Types" Form").
- If the tag XF_<code of the corresponding accounting section>=<>; is not set, the tag XF=<>; in the *XF Configuration* field of the GL Type form is used (see the sections "Off-Balance Accounting Subsystem", "Configuring the "GL Types" Form").

5.4 Setting up a Payment Order (Transferring an Accumulated Amount to a Standard Account)

A payment order is set up in the cumulative account for recording a fee in the client contract with the "High Precision" marker (in the account specified in the custom fee's *Account Type* field in the example in the figure in the section "Service Settings" and in the figure below,, this is the "Accumulate HP" account).

A payment order transfers the balance from the cumulative "High Precision" account to a regular balance account, and the fee becomes effective.



Setting up a payment order

The template of the standard account is specified as the corresponding account (*Target Template* field) in an order. XF accounts must be specified in the templates of both accounts (see the section "Configuring High Precision Account Templates").

> Note that if a payment order must be used to move an amount with high precision to a "High Precision" account, the order must be set up in an account with the "High Precision" indicator. Otherwise (if the order is set up in the corresponding standard account), the amount being transfer will automatically be rounded to fractional currency units (for example, to two decimal points for dollars).

In the order's *Posting Details* field, tags are set that determine rules for working with the difference between the amount accumulated in the "High Precision" cumulative account and the rounded amount transferred through the system of XF accounts to a standard account:

- PRECISION=<number of decimal places>;
- XF_ROUND;

For "High Precision" accounts with the "Value Date Due" and "Sliding" value of the *Due Type* field, the XF_ROUND; tag cannot be set in the account template and in payment orders used for due normalization (with "Account Due" in the *Date Event* field).

Tags can be set in the following combinations:

- For the difference not to remain in the cumulative "High Precision" account, the tag PRECISION=<maximum number of decimal places>; must be specified in the order; the XF_ROUND; tag doesn't have to be specified.
- For the difference to remain in the cumulative "High precision" account, both tags must be specified.
- Not that if the PRECISION tag is not set in the order, the balance in the cumulative "High precision" account remains regardless of the existence of the XF_ROUND tag in the order (since in this case, an amount with two decimal places will be transferred).

It is recommended to use the same value of the PRECISION tag in the order and in the custom fee.

Example of rounding without the XF_ROUND tag:

Original balance in the Accumulate_HP account = –12.13728

When an order is activated, two macrotransactions are generated:

Debit: HP Fee Active – Credit: Accumulate_HP. 12.13728

Debit: Merchant Current – Credit: (Bank) Revenue 12.14

In this case, the balance in the High Precision account is fully debited and the value is rounded for balance accounts. Rounding rules are regulated using the additional standard ROUND tag.

Example of rounding with the XF_ROUND tag:

Original balance in the Accumulate_HP account = –12.13728

When an order is activated, two macrotransactions are generated:

Debit: HP Fee Active – Credit: Accumulate_HP. 12.14 (that is, a "tail" of 0.00272 remains in the High Precision Fee account)

Debit: Merchant Current – Credit: (Bank) Revenue 12.14

When rounding, the cumulative "Accumulate_HP" account balance can change the sign if the rounded value is greater than the original value, see figures below.

Accumulate_HP USD	M	USD	0.00	
Accumulate_HP	M	CZK	0.005	

Balance before activation of an order

Accumulate_HP USD	M	USD	0.00	
Accumulate_HP	M	CZK	-0.005	

Change in sign after an order has been activated when rounding up

When a fee is accumulated in a "High Precision" account in one currency and then transferred by an order to a standard account in another currency, the accumulated amount first goes through an XF account and is rounded to the currency's fractional units (for example, two decimal places for USD).

5.5 Other Options for Working with "High Precision" Accounts

The functionality shown below should not be used without consulting OpenWay representatives and careful testing:

- Limit normalization between high precision accounts.
- Due normalization between high precision accounts.

> If the account to which funds are transferred (Due Account) from a "High Precision" cumulative account is a regular (balance) account, rounding is performed as follows:
>
> - If the XF_ROUND; tag is set in the template of the cumulative "High Precision" account, a rounded amount is transferred between "High Precision" accounts (cumulative account – >XF account) (same as how the XF_ROUND tag works in a payment order, see the section "Setting up a Payment Order (Transferring an Accumulated Amount to a Standard Account)"). In this case, a balance ("tail") remains in the cumulative "High Precision" account. A rounded amount (standard rounding) is transferred between balance accounts (XF account ->Due Account).
> - If the XF_ROUND tag is not set, the full amount (with the number of decimal places specified by the PRECISION tag) is transferred between "High Precision" accounts (cumulative account ->XF account). In this case, the balance in the "High Precision" cumulative account is debited in full. A rounded amount (standard rounding) is transferred between balance accounts (XF account ->Due Account).
>
> For "High Precision" accounts with the "Value Date Due" and "Sliding" value of the *Due Type* field, the XF_ROUND; tag cannot be set in the account template and in payment orders used for due normalization (with "Account Due" in the *Date Event* field).

- High precision interest accrual. The High Precision indicator must be specified in the template of the interest account and the tag "PRECISION=<number of decimal places>"; must be set in the

Template Details field. The revenue/expense account participating in the interest accrual entry must also be a "High Precision" account.

> If the PRECISION tag is not set in the account template, the default value specified in the "GL Types" form for the account type "High precision account" is used (Full → Configuration Setup → Accounting Setup → GL Types), see the section "Configuring the "GL Types" Form". The default value of the PRECISION parameter is "6" (that is, by default, the tag PRECISION=6; is set in the "GL Types" form).
>
> If the PRECISION tag is also not set in the "GL Types" form, this tag's maximum value is used – "10".
>
> If an entry is made between two "High precision" accounts that have different PRECISION tag values, rounding is performed to the lesser value.

If the PRECISION tag is not set in the account template, the default value specified in the "GL Types" form for the account type "High precision account" is used (Full → Configuration Setup → Accounting Setup → GL Types), see the section "Configuring the "GL Types" Form". The default value of the PRECISION parameter is "6" (that is, by default, the tag PRECISION=6; is set in the "GL Types" form).
If the PRECISION tag is also not set in the "GL Types" form, this tag's maximum value is used – "10".
If an entry is made between two "High precision" accounts that have different PRECISION tag values, rounding is performed to the lesser value.

5.6 Generating FX Entries

- When transferring funds from a "High Precision" account to a balance account through XF accounts, if the account currencies differ, FX entries are not generated by default. Funds are transferred through XF accounts, and an XF account is searched for in the same currency as the account for which this search is being made. "Hidden" conversion is performed – an entry is made in balance accounts for the amount in the currency of balance accounts (with conversion at the "Middle" rate).
- To generate FX entries and convert according to the rate defined in the Service, in the *XF Configuration* field of the "GL Type" form set the tag FX_CONV=<code of the corresponding accounting section1>, <code of the corresponding accounting section2> according to which FX entries must be made (see the section "Configuring the "GL Types" Form"). In this example:
- For balance accounting ("Balance" in the *Name* field), set the tag FX_CONV=h;.
- For the "High Precision" section – FX_CONV=B;.