WAY4™ Accounting Schemes

Contents

Introduction

This document is intended for bank or processing centre 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 reference material from WAY4 documentation series:

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

The following conventions are used throughout this document:

- Field labels in screen forms are typed in *italics*.
- Button labels used in screen forms are placed in square brackets, such as [Approve].
- Menu selection sequences are shown with the use of arrows, such as Configuration Setup → Contract Types.
- Item selection sequences, in the system menu, are shown with the use of different arrows, such as Database => Change password.
- The  sign warns that there is an increased chance of making an incorrect action.
- Messages marked with the  sign contain information about important features, additional facilities, or the optimal use of certain functions of the system.

Chapter 1. Terms and Definitions

WAY4 is used to:

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

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 a contract: card contract (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 a natural person, a legal person, or a bank branch)
- Service Package – contains a list of transactions, fee parameters, transaction processing rules, and usage limiters
- Accounting Scheme – determines a contract's accounts, their properties and relations between them (see Fig. 1)

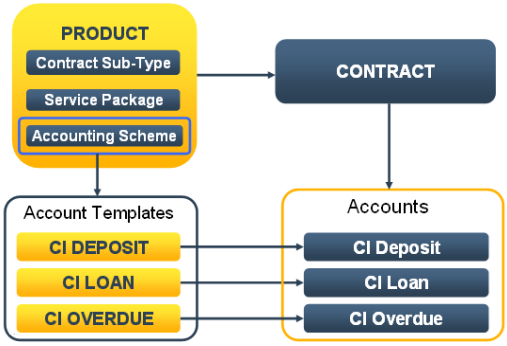


Fig. 1. Relations between system objects

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

Chapter 2. Accounting Schemes

An Accounting 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 details see the document "WAY4 Accounting".

The system allows the following accounting functions to be executed through account template configuration in an Accounting Scheme:

- Accrual of interest on account balances (see "Interest Properties")
- Amount normalisation (see "Limit Normalisation") – process where funds are automatically redistributed among the contract's accounts when their balances change
- Due normalisation (see "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 "GL Properties")

Form "Account Schemes"

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

This will open the "Account Schemes" grid form (see Fig. 2).

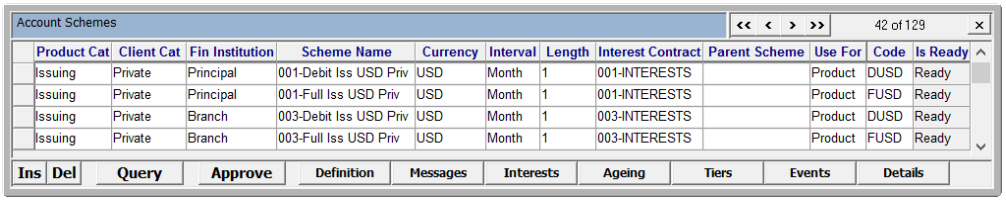


Fig. 2. Form for entering and viewing Accounting Scheme data

The "Account Scheme" form is used to access the complete list of registered Accounting 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 Accounting Scheme, it is more convenient to use special forms with preliminary data filtering (see "Special Forms for Working with Accounting Schemes").

The "Account Scheme" form contains fields for specifying main parameters of an Accounting Scheme. For a description of the fields, see "Main Accounting Scheme Parameters".

The form also contains the following control buttons:

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

Main Accounting Scheme Parameters

Product Cat

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

Client Cat

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

Fin Institution

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

Scheme Name

The name of an Accounting 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).

Parent Scheme

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

Used For

There are two types of Accounting Scheme:

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

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.

Interval

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

Length

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

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 Accounting Scheme.

Interest Contract

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

Is Ready

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

- The field contains the "Ready" value if changes made to the Accounting Scheme have been approved.
- The field contains the "Not Ready" value if changes made to the Accounting Scheme have not been approved.

For information about Accounting Scheme approval, see "Accounting Scheme Approval".

Additional Parameters of an Accounting Scheme

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

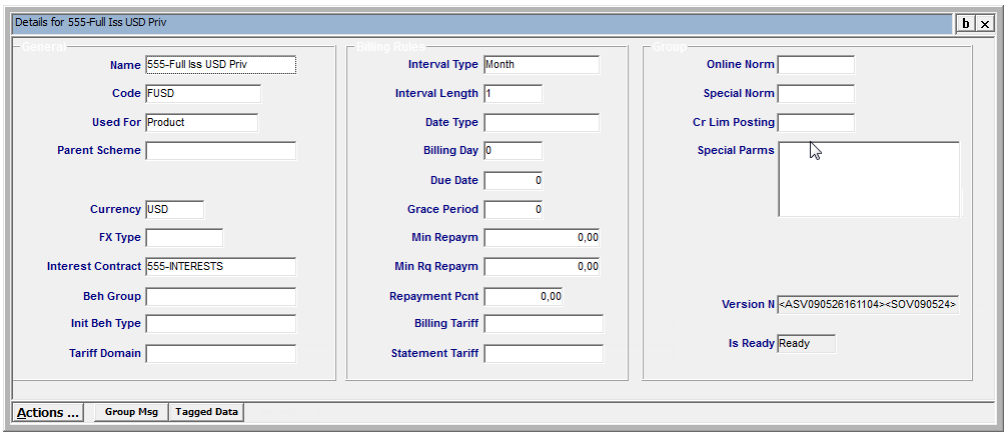


Fig. 3. Form "Details for <name of Accounting Scheme>

The form with additional Accounting Scheme data contains the following controls:

- Clicking the [Actions] button opens the following context menu items:
  - [Check] – verify Accounting Scheme parameters (see "Checking Accounting Schemes").
  - [Duplicate] – copy the Accounting Scheme (see "Copying Accounting Schemes").
  - [Save Tagged Data] – save the tags entered in the *Special Parms* field (see "Entering and Editing Tags").
- The [Group Msg] button is used to set up client message templates (see "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 "Entering and Editing Tags").
- The [Included] button is used to work with additional Accounting Schemes. For more information, see the sections "Used For" and "Working with Additional Accounting Schemes (Included)".

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

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

FX Type

Conversion rate type. For more details on currency conversion in WAY4, see the Currency Conversion Administrator Manual.

Interest Scheme

This parameter is used to calculate the daily interest rate using an annual interest rate. The *Interest Scheme* parameter in an Accounting Scheme redefines the value of the parameter with the same name in a financial institution. The *Interest Scheme* parameter in an Accounting 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 determine the daily interest rate, the length of a year will depend on the value of the USE_MONTH_WEIGHT global parameter (see the document WAY4™ Global Parameters and sections "Number of Days in a Year", "Determining a Daily Interest Rate" in 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 sections "Number of Days in a Year", "Determining a Daily Interest Rate" in 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 sections "Number of Days in a Year", "Determining a Daily Interest Rate" in the document "Interest Accrual".

Behavior Group

The group of possible Behaviour Types for contracts using this Accounting Scheme.

Init Beh Type

The Behaviour Type assigned at the time of registering a contract using this Accounting Scheme.

Tariff Domain

For each Accounting 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 Advanced Tariff Management module is not included in the basic WAY4 configuration and is supplied under an additional agreement with the WAY4 vendor.

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 "Business Calendar" section in the document "WAY4™ Dictionaries").

  For example, the *Interval* field is set to "Month", 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 in 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 *Billing Day* 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 Monday, 02 April, and if the *Billing Day* field is set to "4", the actual last day of the billing cycle will be Thursday, 05 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.

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").

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.

For example, an Accounting Scheme has the following parameters:

- A billing cycle is one month.
- The *Due Date* field is set to "16".
- The Accounting 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 normalisation 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 normalisation, i.e. in our example funds are transferred from the "Cl Paym Due" account to the "Cl OVD" account at the end of the 20th day.

Grace Period

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

Min Repaym

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

Min Rq Repaym

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

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 delivery 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 Advanced Tariff Management module is not included in the WAY4 basic configuration and is supplied under an additional agreement with the WAY4 vendor.

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 delivery 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 supplied under an additional agreement with the WAY4 vendor.

Online Norm

The field is used to specify the order of normalisation of accounts created using this Accounting 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 WAY4™ Global Parameters Administrator Manual), normalisation 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 normalisation between accounts of this Accounting Scheme is performed immediately during macrotransaction posting.
  - If the *Online Norm* field contains the value "No", account normalisation is not performed during macrotransaction posting; in this case, a special DB Manager user menu item performing contract account normalisation may be used when necessary (e.g. before opening the next banking day). Account normalisation can also be performed during approval of a contract or the corresponding Accounting Scheme.

Special Norm

The *Special Norm* field is used in indirect multicurrency normalization (see the section "Limit Normalisation"). The field makes it possible to redefine the global parameter MULTICURRENCY_NORMALIZATION (see the WAY4™ Global Parameters Administrator Manual), in a specific Accounting Scheme.

When it is necessary to set up multicurrency account normalisation 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, multicurrency normalisation 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, multicurrency normalisation for this Scheme's accounts can be performed either using standing payment orders (see section "Multicurrency Normalisation" in the Standing Payment Orders Adminstrator Manual) or through indirect multicurrency normalisation (see "Limit Normalisation").
  - If the *Special Norm* field contains the "None" value, multicurrency normalisation is not performed.

When performing multicurrency normalization using payment orders, the *Special Norm* parameter is not used.

Cr Lim Posting

This field allows for configuring how credit limits are posted and reflected in accounts of contracts that use this Accounting 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 Accounting 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 Accounting Scheme:
  - When the value of the field is "Yes", credits limits are reflected in accounts based on this Accounting Scheme's templates.
  - When the value of the field is "No", credit limits are not reflected in accounts of this Accounting Scheme.
- If the global parameter CREDIT_LIMIT_POSTING is not set, and the *Cr Lim Posting* field for the Accounting 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 Accounting Scheme is not filled in, nor is it filled in for the financial institution, credit limits are not shown in this Accounting Scheme's accounts.

Special Parms

This field is used to specify additional parameters of Account Schemes as tags. For more details, see the section "Tags Used when Configuring Accounting Schemes and Account Templates".

Special Forms for Working with Accounting Schemes

In most cases, it is convenient to use special forms containing a list of Accounting Schemes selected by financial institution and contract category:

- To view a*nd enter inf*ormation on Accounting Schemes for private clients' issuing contracts, use the "Private Issuing Account Schemes" form (see Fig. 4), opened by selecting "Full → Configuration Setup → Products → Issuing Private Products → Private Issuing Account Schemes" from the user menu.
- To view and enter information on Accounting 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 enter information on Accounting Schemes for acquiring contracts, use the "Acquiring Account Schemes" form (Full → Configuration Setup → Products → Acquiring Products → Acquiring Account Schemes).
- To view and enter information on Accounting Schemes for bank contracts, use the "Bank Account Schemes" form (Full → Configuration Setup → Accounting Setup → Bank Account Schemes).

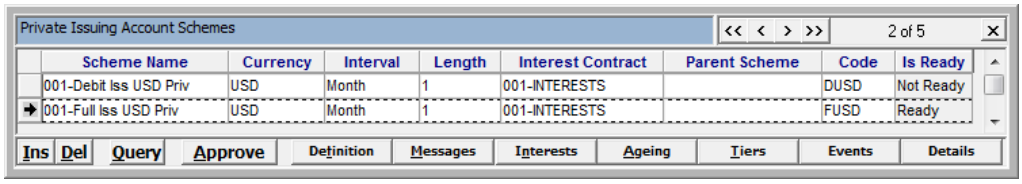


Fig. 4. Form for entering and editing Accounting Scheme data for issuing contracts of private clients

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

Chapter 3. Configuring Account Templates

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

Account Types

"Account Types" Form

An account template description in an Accounting 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).

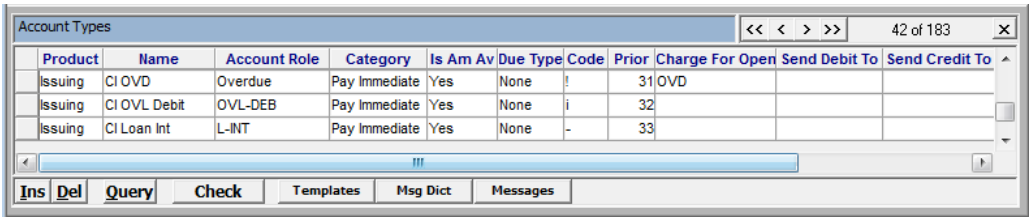


Fig. 5. Form for registering and configuring account types

The "Account Types" form (see Fig. 5) 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 are created ("Loan Retail" and "Loan Cash") with the "Loan" role for use within 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 some features of processing of accounts of this type; the list of account categories is predetermined in the system and contains the following values:
  - "Pers Limit" – when lower limit normalisation is performed for an account of this category, the threshold value will be the contract's credit limit, not the value of the *Low Limit Amount* field of the account template (see examples in section "Limit Normalisation Mechanism"). This category is assigned to loan accounts.
  - "Cr. Lim Payment Due" – this category is used to set up account templates for keeping amounts made due as part of credit limit. In this case, when contract normalisation 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 summed up and compared with the contract's credit limit; therefore, when funds are transferred from the loan account to the due account, limit normalisation (OVL → Cl Loan) is not performed for the loan account (see example 2 in section "Limit Normalisation Mechanism").
  - "Cr. Lim Overdue" – this category is used for overdue loan accounts ("Cl OVD") in the same way as the "Cr. Lim Payment Due" category.
  - "Payment Due" – when a card statement is generated, balances of accounts with this category are summed up to show the due amount.
  - "Pay Immediate" – when a card statement is generated, balances of accounts with this category are summed up 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 summed up, 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 normalisation.

    Accounts with the "Shared Limit" category implement an obsolete type of account behaviour. 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 *Interest Algorithm* field). 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 normalisation type; for more details on this field's values, see "Ageing". During account template setup, a due normalisation 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 Accounting 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 within the same Product category. Note that the system differentiates between uppercase and lowercase characters in this field.

  ⚠ In the standard system setup, 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 within the Accounting Scheme. Priority values affect the order of interest accrual in accounts and repayment of loan account balances. When a new template is added in an Accounting Scheme, a 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.

  ℹ This field is left for compatibility for earlier system versions. It is not recommended that it be used.
- Fields *Send Debit To* and *Send Credit To* are used in the system to describe a pair of active and passive accounts: for a passive account, the *Send Debit To* field contains an active account used to debit the current account.

Correspondingly, the *Send Credit To* field for an active account contains a passive account 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 Accounting Scheme was not approved, an error message is shown after changes to the account type. The [Messages] button is used to get information about Accounting 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 Accounting Schemes that must be approved. After a scheme is appoved, changes to contracts must be applies 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.

Classification of Account Types (Account Roles)

Account types are classified into groups depending on their use (depending on account role), for example "Deposit", "Loan", 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 Fig. 6.

 If new records must be added to the handbook, contact WAY4 vendor representatives.

Roles groups allow the following:

- Systematization of contract accounts for their clearer reflection in the interface.
- Additional classification of 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 with 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)

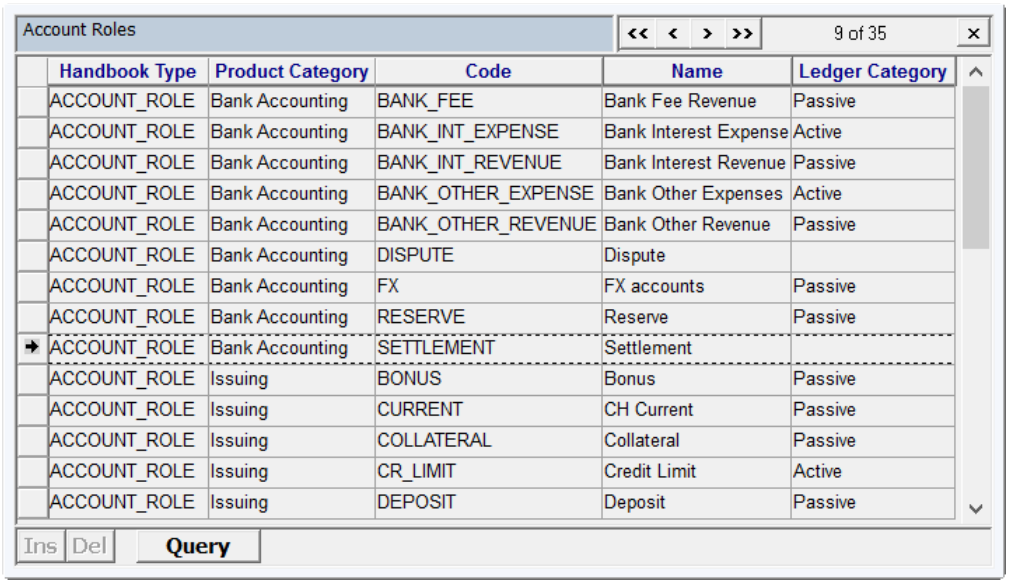


Fig. 6. "Account Roles" form

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 the account group belongs to a particular Product category.

  - "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 Table 1). For example, account types with the "Dispute" role (in the handbook, "Dispute" belongs to the "Bank Accounting" category) can be used in bank issuing and acquiring Products (Accounting Schemes). I.e. 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 on client loan contract accounts (for group portfolio reserves, accrued reserves will be shown on bank contract accounts).

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

Table 1. Account role groups

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Proposed 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 private and corporate clients, 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 Accounting Schemes.
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 Accounting Schemes.  This group is similar to the "Credit Limit" group, but put in a separate group to get nonoverlapping balances on the role group level.

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Proposed use**
Issuing	Loan	Loan debt	Credit cards, non-card Products (including all technical sections 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 (unsanctioned overdraft)	Credit cards
Issuing	Fee	Fee (client obligation)	Credit cards, non-card credit products
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
Issuing	Penalty	Fines and penalties	Credit cards, non-card credit products
Issuing	Guarantee	Guarantees	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 reevaluation due to depreciation)

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Proposed use**
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 Accounting 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 client)	
Technical accounts			

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Proposed use**
Issuing	Technical	Technical account	Used in various Products, of any category to record funds on off-balance technical accounts (on 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 for 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

Product category in the "Account Roles" handbook	**Role name**	**Description**	**Proposed use**
Bank accounts (Settlements)			
Bank Accounting	Settlement	Correspondent account (full or partial mirror of the account in the ABS – for example, entries for crediting a correspondent account can be made in the ABS and not reflected in WAY4).	Settlements with IPS, with 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	Accounts of ATMs, Cash-in devices, virtual teller contracts, Main contracts of Cash POS.
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 kept in WAY4.
Dispute accounts			
Acquiring/ Issuing	Dispute	Account for reflecting amounts for which dispute work is in process, in client schemes, payment system schemes, inter-branch settlements	Dispute operations until resolution.

Configuring the "GL Types" Form

Various accounting sections are configured in the "GL Types" form (Full → Configuration Setup → Accounting Setup → GL Types), see Fig. 7. Settings in this form for a certain accounting section are used for all account templates of the specified type. Settings can be redefined for a specific account template.

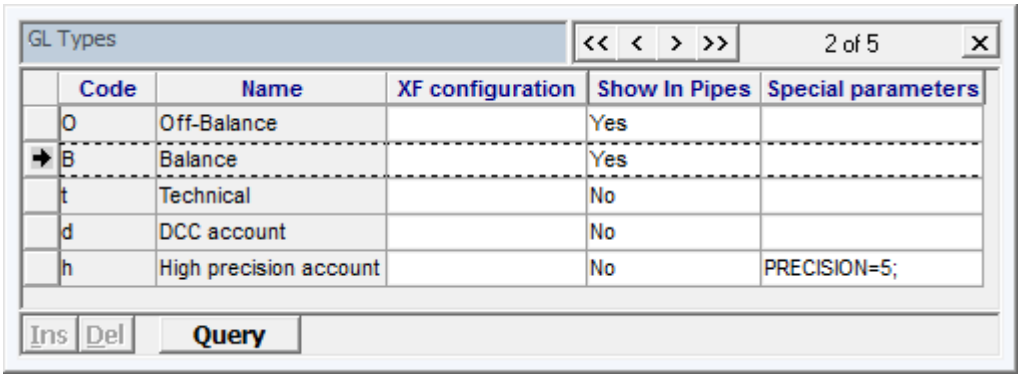


Fig. 7. "GL Types" form

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=< >;. 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 accounting sections (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 (conditional on 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 (conditional on 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".

Full Information about Accounting Scheme Templates

For more detailed information about each template in an Accounting Scheme, click the [Definition] button in the "Account Schemes" form (see Fig. 2 in section "Form "Account Schemes"") or in a special form for working with Accounting Schemes (see Fig. 4 in section "Special Forms for Working with Accounting Schemes"). As a result, the "Definition for <name of Accounting Scheme>" form will be opened. See an example of the form for private clients' issuing contracts in Fig. 8.

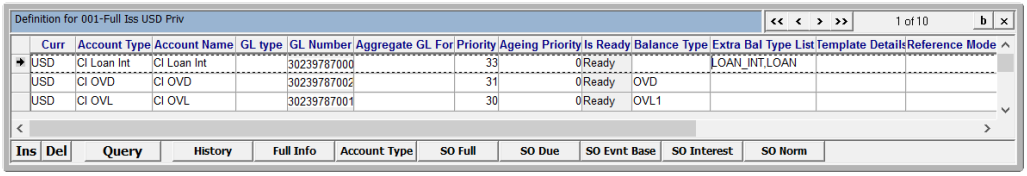


Fig. 8. Form for accessing account templates

In the "Definition for <name of Accounting Scheme>" form that opens, click the [Full Info] button. As a result, the "Full Info for <name of account template>" form will be displayed. This is the most convenient form for registering new account templates or modifying old ones (see Fig. 9).

Fields used in special forms for working with Accounting Schemes of various contract categories (see "Special Forms for Working with Accounting 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 Accounting Scheme data for bank contracts (Full → Configuration Setup → Accounting Setup → Bank Account Schemes). This form contains the *Fin Institution* and *HeadOffice GL#* fields, used to set up interbranch routing (see section "Interbranch Transactions" in the Financial Institutions Administrator Manual) and absent from the account template full information form.

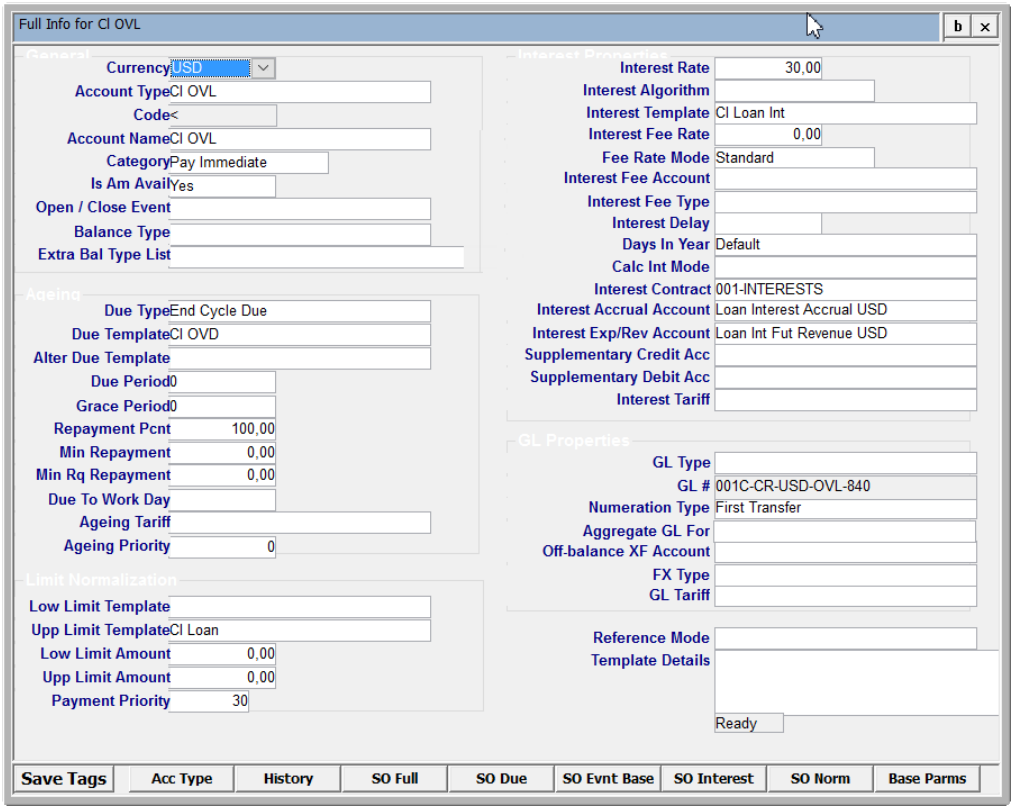


Fig. 9. Form for viewing and entering information about Accounting Scheme accounts

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 details on standing payment orders, see the Standing Payment Orders Administrator Manual.

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

The [Tagged Data] button is used to optimise work with the *Template Details* field. Clicking this button opens the grid form for entering and editing tags in the *Template Details* field (see "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 normalisation parameters
- "Limit Normalisation" – amount normalisation parameters
- "Interest Properties" – interest accrual parameters

- "GL Properties" – GL parameters

General

This group of fields in the "Full Info for <name of account template>" form (see Fig. 9) is used to enter main account information.

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

  ⚠ An Accounting Scheme may not contain two accounts templates of the same type in the same currency. If it is necessary to add an account templates whose properties are the same as the properties of another account template in the Accounting Scheme, add a new account type (see "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 Accounting Schemes.

  When executing transactions on an account with this marker, 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 *Is Am Avail*="No", when a transaction on the account is made, the amount available is calculated based on this balance type's value.

  Open/Close Event – name of the type of the Event opened when funds appear in the account and closed when the account balance becomes zero.

  ⚠ An exception is the situation when a macrotransaction is posted leading to a certain Event being closed due to clearing of the account balance, another Event opens in connection with the formation of balances on accounts or balances (an Event related to an account template and/or balance type). It is possible for these Events to not be linked to each other (for example, by an Event package). In this case, the Event only closes if the NOT_USED_IN_CHAIN tag is specified in the Event type. Otherwise, the Event will not close.
- *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" in the Balance Types Administrator Manual).
- *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".

Ageing

This group of fields in the "Full Info for <name of account template>" form (see Fig. 9) is used to determine rules for funds transfer during due normalisation.

When a banking day is opened (see section "Start of Day Procedure" in the Daily Procedures User Manual), 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 normalisation rules.

- Field *Due Type* – due normalisation type; the value of this field determines how the due date is calculated:
  - "Value Date Due" – the normalisation 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 section "Posting" in the WAY4™ Service Packages Administrator Manual).

    If the *Value Days* field is not filled in for a Service (equal to zero) and a "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).
  - The system differentiates between business days and days off using a business calendar (see the "Business Calendar" section in the WAY4™ Dictionaries Administrator Manual); this value is used in acquiring Accounting Schemes.
  - "End Cycle Due" – the normalisation date is determined by the end of the corresponding billing cycle according to the value of the *Due Period* field ("0" – normalisation at the end of the current billing cycle; "1" – normalisation at the end of every second billing cycle staring with the account opening date; "2" – normalisation at the end of every third billing cycle staring with the account opening date, etc. In this case, an account is considered opened when a non-zero balance appears in the account). The normalisation amount is the account's incoming balance on the normalisation date; macrotransaction posting is affected by the value of the POST_DUE global parameter (see the WAY4™ Global Parameters Administrator Manual).
  - "Contract Due" – the normalisation date is determined based on calculated contract functional dates. Any functional date can be used to post due normalisation. 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 normalisation, 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 Accounting Scheme's) calendar, and for the financial institution every day, including weekends/holidays is opened when the "Contracts - Daily Update" procedure is performed, due normalisation will be performed when opening/closing (according to the global parameter PAYMENT_DUE_ADVANCE or the same tag in the Accounting Scheme) the first working day after the calculated date, according to the calendar of the financial institution/Accounting Scheme.

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 normalisation date is calculated by adding to the beginning of a billing cycle the number of calendar days specified in the *Due Period* field. The normalisation amount is the account's incoming balance on the normalisation date; macrotransaction posting is affected by the value of the PAYMENT_DUE_ADVANCE global parameter (see the WAY4™ Global Parameters).

  If date schemes are used to define contract functional dates, it is recommended to use the "Contract Due" normalisation type instead of "Payment Due" when configuring normalisation.

- "Quarter" – the normalisation 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 normalisation amount is the account's incoming balance on the normalisation 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 normalisation 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 shifter (for example, when the *Date Type* field value is "From Open Date"), normalisation may be performed not on the first of the month, but at the end.

- "Long Payment Due" – this value is left for compatibility with earlier system versions. Works in the same way as the "Payment Due" value.

- "Sliding" – used to set a normalisation period (in calendar days) when the normalisation date is calculated by adding the normalisation period (the value of the *Due Period* field) to the date when funds are transferred to the account. The normalisation amount is the account's incoming balance on the normalisation date.

  Example 1:

  An Accounting Scheme contains templates "Cl Loan" and "Cl OVD". A loan must be repaid within 45 days after a transaction is performed: *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 performed. As a result, the balance of the "Cl Loan" account became -40 euro.
  - On 05 April, a card transaction for 60 euro was performed. 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 performed 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 performed 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 Accounting Scheme contains templates "Cl Loan" and "Cl OVD". A loan must be repaid within 45 days after a transaction is performed: *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 performed. As a result, the balance of the "Cl Loan" account became -40 euro.
  - On 06 March, a card transaction for 60 euro was performed. 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 performed on 05 March and partially repaid the amount due for the transaction performed on 06 March.
  - On 20 April, i.e. 45 days after 06 March, the unpaid part of the amount due for the transaction performed on 06 March (-50 euro) will be transferred to the overdue loan account "Cl OVD".

- "Sliding+Clear" – the normalisation date is calculated by adding the normalisation 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

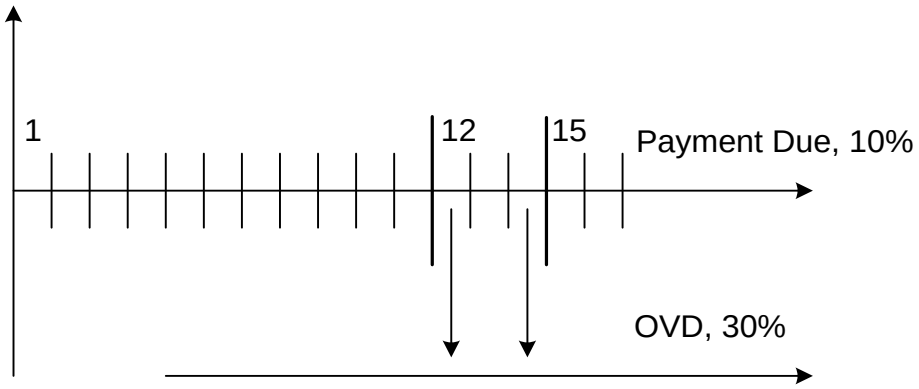
normalisation amount is the account's incoming balance on the normalisation date.

- "Fixed Day Due" – the normalisation date is the calendar day after the day of the month specified in the *Due Period* field.

- 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 the *Due Template*, and the remaining funds are transferred to the *Alter Due Template*.
- *Due Period* – the value of this field is used considering the normalisation type (the value of the *Due Type* field) to determine the normalisation period, i.e. 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 normalisation dates. In this case, a normalisation 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.

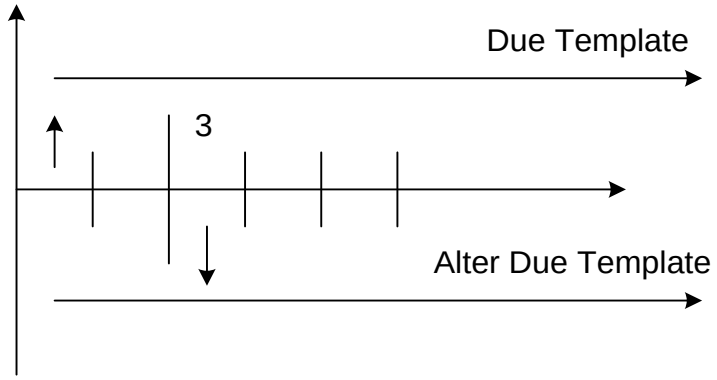
    For instance, assume that *Due Period* is "15", *Grace Period* is "4", and funds were transferred to the account on the first day of the month. At the end of the due period, funds are transferred from the due account to the overdue account. The loan interest rate in the due loan account is 10%; the loan 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 (i.e. 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 the *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 "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 *Min Repayment* field, the system will use the opposite (positive) value as the value of the *Min Rq 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 *Min Rq Repayment* field will be transferred to the account specified in the *Due Template* 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 the system. If the bank's distribution package does not include the WAY4 Tariffs module, the field is empty. See the section "Tariffs with the "Ageing" Role" of the document "WAY4™ Advanced Tariff Management".

  When "Contract Due" is specified in the *Due Type* field, the *Ageing Tariff* field is not filled in. In this case, a normalisation date is set using contract functional dates and cannot be redefined with an "Ageing" tariff.

The WAY4 Tariffs module is not included in the basic WAY4 configuration and is supplied under a separate agreement with the WAY4 system vendor.

- *Ageing Priority* – this field affects the due normalization process for "End Cycle Due" and "Quarter" account types.. First, due normalization is performed for accounts with a higher priority. The value of the *Payment Priority* field is considered. I.e., if there are two accounts: Account 1 with an *Ageing Priority* field value of "3" and *Payment Priority* field value of "2", and Account 2 with an *Ageing Priority* field value of "3" and *Payment Priority* field value of "1", due normalization is first performed for Account 1.

Limit Normalisation

Description of Fields

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

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 Normalisation" group of fields in the "Full Info for <name of account template>" form (see Fig. 10):

- *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 Accounting Scheme. Priority values affect the order of interest accrual in accounts and repayment of loan account balances. When a new template is added in an Accounting 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.

  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.

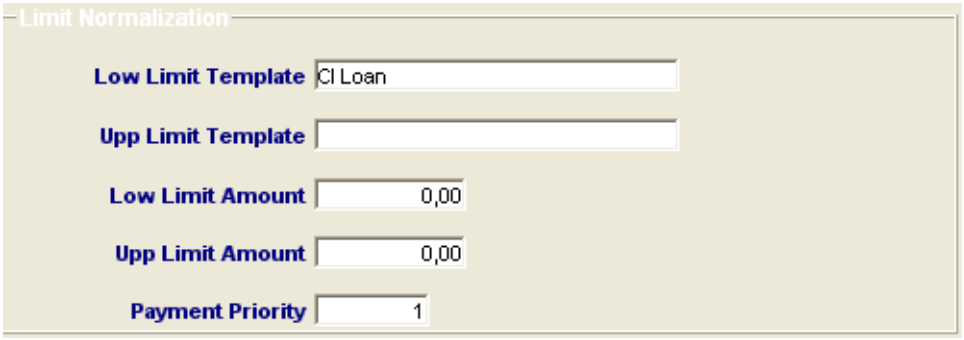


Fig. 10. Parameters of setup of limit normalisation of a deposit account

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

To approve changes made in a template, click the [Approve] button in the "Account Schemes" form (see Fig. 2 in section "Form "Account Schemes"").

Limit Normalisation Mechanism

The limit normalisation 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 USD 1,000 credit limit is set for a contract. The balance of the loan account ("Cl Loan") is USD -1,000.
- 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 normalisation rules. Therefore, the balance of the "Cl Loan" account becomes USD -900, and the balance of the "Cl Paym Due" account becomes USD -100.

Example 2.

- A USD 1,000 credit limit is set for a contract. The balance of the loanf account (Cl Loan) is USD 900.
- A card transaction for the amount of USD 100 is performed.
- With this setup, the balance of the "Cl Loan" account does not exceed the contract's credit limit amount (900<1000); as a result, the card transaction is reflected in the "Cl Loan" account (the balance of the "Cl Loan" account becomes USD -1,000), lower limit normalisation is not performed for the "Cl Loan" account.

Example 3.

- A USD 1,000 credit limit is set for a contract. The balance of the loan account ("Cl Loan") is USD -1,000.
- 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 normalisation rules. Therefore, the balance of the "Cl Loan" account becomes USD -900, and the balance of the "Cl Paym Due" account becomes USD -100.
- A card transaction for the amount of USD 100 is performed.
- 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=1000); as a result, lower limit normalisation 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 USD -100).

- 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* value 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 normalisation is indirect multicurrency normalisation. This mechanism is used if an Accounting Scheme contains accounts in different currencies.

The standard mechanism of amount normalisation between accounts in different currencies uses standing payment orders of a special type (see section "Multicurrency Normalisation" in the Standing Payment Orders Administrator Manual).

Indirect multicurrency normalisation (see Fig. 11) is used when there are no standing payment orders and the MULTICURRENCY_NORMALIZATION global parameter is set to "Y" (see the WAY4™ Global Parameters Administrator Manual).

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

Indirect multicurrency normalisation is possible in Accounting Schemes with more than two currencies. For example, an Accounting Scheme may use currencies EUR, USD and JPY. In this case, all three accounts participate in normalisation, 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 Accounting Scheme whose main currency is EUR contains two accounts – Cl Deposit (USD) and Cl Deposit (EUR).

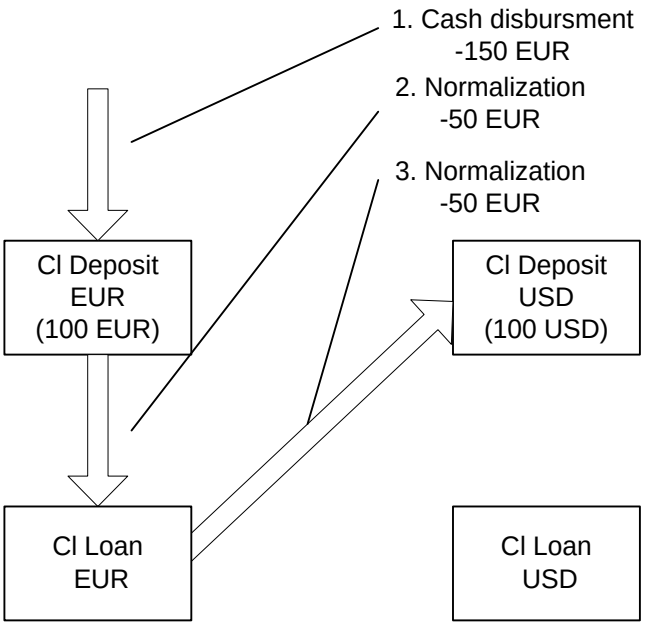


Fig. 11. Multicurrency normalisation

In case of indirect multicurrency normalisation (see Fig. 11), when a transaction is performed, the amount is first debited to 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 normalisation of an account also affects the contract's accounts of the same type registered in other currencies.

During multicurrency normalisation, the debited amount including the fee and rounding may not exceed the amount of the balance in the account being debited.

Interest Properties

This group of fields of the "Full Info for <name of account template> form (see Fig. 9) is used to specify interest accrual conditions.

- *Interest Rate* – interest rate

  When it is necessary to change the interest rate of an account for 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 Accounting Scheme form.
  - The new interest rate will be used after execution of the "Apply Account Scheme Changes" procedure within the Contracts Daily Update procedure, meaning it will become effective the day after the changes are made.

    The "Contracts Daily Update" procedure is executed automatically within the procedure for opening a new banking day.

  Execution of the "Apply Account Scheme Changes" procedure may take significant time. To shorten the time it takes to execute the Contract Daily Update procedure, the "Apply Account Scheme Changes" procedure can be executed separately using the menu item "Full → Configuration Setup → Products → Apply Account Scheme Changes". This procedure should be executed after processing the last macrotransaction for the current day.

- *Interest Algorithm* – interest accrual algorithm. For more details on the values of this field, see section "Interest Algorithm" in the Interest Accrual Administrator Manual.
- *Interest Template* – account to which interest is accrued.
- *Interest Fee Rate* – rate of the fee charged in the following cases:
  - Charge of a fee from account interest income; us ed for automatic collection of income tax on deposit interest
  - Accrual of penalties

  For more details, see section "Interest Fee Rate" in the Interest Accrual Administrator Manual.

Previously, the *Interest Fee Rate* field was used as a basic functionality for setup of the material gain tax. Currently, it is recommended that charge of this tax be set up using the WAY4 Tariffs module. The module is supplied under a separate agreement with the WAY4™ system vendor.

- *Fee Rate Mode* – drop-down list of fee charge algorithms. For more details, see section "Interest Fee Rate" in the Interest Accrual Administrator Manual.
- *Interest Fee Account* – account to which fees will be transferred.
- *Interest Fee Type* – drop-down list of fee types charged according to the miscellaneous service set up in the contract's Service Package (see section "Miscellaneous Services" in the WAY4™ Service Packages Administrator Manual). The fee is charged to the client after the account interest accrual procedure is executed.
- *Interest Delay* – determines the date starting with which interest is accrued and the date up to which interest is accrued. For more details, see section "Interest Delay" in the Interest Accrual Administrator Manual.
- *Calc Int Mode* – allows loan interest to be accrued on the day when the loan is repaid. For more details, see section "Calc When Credit" in the Interest Accrual Administrator Manual.
- *Days In Year* – used to calculate a daily interest rate using an annual interest rate. For more details, see section "Number of Days in a Year" in the Interest Accrual Administrator Manual.
- *Interest Contract* – bank contract whose accounts are used during interest accrual. For more details, see section "Setup of Parameters in Account Templates" in the Interest Accrual Administrator Manual.
- *Interest Accrual Account*, *Interest Exp/Rev Account* – bank accounts used during interest accrual. For more details, see the "Interest Accrual" Administrator Manual.
- *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" in the document "Interest Accrual".
- *Interest Tariff* – drop-down list of tariffs with the "Interest" role registered in the system. If the bank's distribution package does not include the WAY4 Tariffs module, the field is empty.

  The WAY4 Tariffs module is not included in the basic WAY4 configuration and is supplied under a separate agreement with the WAY4 system vendor.

GL Properties

This group of fields of the "Full Info for <name of account template> form (see Fig. 9) is used to determine how an account interacts with the General Ledger.

For more details, see the document "WAY4 Accounting".

- *GL#* – WAY4 GL account number:
  - The account number can be selected from a drop-down list. 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 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 particulars of data storage and does not influence the correctness of posting transactions to GL accounts.

  When an Accounting Scheme's *Parent Scheme* field is filled in, the *GL#* and *HeadOffice GL#* fields will not be copied from the parent Accounting Scheme to the child Scheme if these fields are already filled in the child Accounting Scheme. Moreover:
  - If the *GL#* and *HeadOffice GL#* fields in the child Accounting Scheme's template are not filled in, values are inherited from the parent Accounting Scheme the first time the Scheme is approved.
  - When the *GL#* and *HeadOffice GL#* fields in the child Accounting 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 Accounting Scheme are changed, the new values are not inherited to the child Accounting Scheme.
  - If the child Accounting Scheme's *GL#* and *HeadOffice GL#* fields are not filled in, on repeat [Approve], values are inherited from the parent Accounting 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 the account belongs to the balance subsystem of accounting.
- "Off Balance" – specifies the account belongs to the off-balance subsystem of accounting.
- "Technical" – technical accounts used to support Product logic 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 Accounting 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, the check is not made.

  The *GL#* field is filtered by the value of the *GL Type* field for new Accounting Schemes, when adding a new account template. For existing Accounting 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 Transfer" – a subsidiary GL account number is generated when the first posting is made to it according to a custom procedure.

    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 normalisations
  - "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.

  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 has a "Y" (Yes) value, 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.

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 Accounting 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 Accounting Scheme is approved if it is approved after a contract).


Setup of a custom account numeration procedure is not included in the standard system setup and is supplied by the WAY4 system vendor under a separate agreement.

- *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.
  - "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 Currency Conversion Administrator Manual.
- *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 base configuration and is supplied by separate agreement with the WAY4 vendor.

- *Reference Mode* – used to configure synchronisation of the main Scheme's templates with the templates of an additional Accounting Scheme. For more information, see the section "Working with Additional Accounting Schemes (Included)".
- *Template Details* – this field is used to specify additional parameters of the account template as tags; the field is used to support the *Off-balance acco*unting (see "Off-Balance Accounting Subsystem").

Off-Balance Accounting Subsystem

Data in both balance and off-balance subsystems is closed, i.e. 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 (e.g. 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 Accounting Scheme in WAY4:

- Accounting subsystem (balance, off-balance) to which the account belongs. The *Aggregate GL For* field (until version 03.35.30 the *Use GL#* field) of an account template is used for this. The field can take on one of four values: "Sub GL" (until version 03.35.30, "For Analytic"), "Consolidated GL" (until version 03.35.30, "For Synthetic") – the account belongs to the balance subsystem; "Off-Balance Sub GL" (until version 03.35.30, "For Off-Balance Analytic"), "Off-Balance Consolidated GL" (until version 03.35.30, "For Off-Balance Synthetic") – 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>;"

    where:

Bal_N is the code of the type of the balance account of a bank contract

Off_N is the code of the type of the off-balance account of a card contract

<Bal_Default> is the code of the type of the balance account of the default bank contract

⚠ Only one of the above fields may be filled 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 (e.g. when different income accounts must be used), the *Template Details* field is filled in.

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

Example 1:

When a contract goes from behaviour type 1 to behaviour 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 corresponding off-balance account → crediting the "Cl Loan Int Off Bal" account. The corresponding 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 behaviour type 1 to behaviour 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 behaviour type 1 to behaviour 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, "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 2 are registered in the system.

Table 2. List of registered accounts that shows 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 behaviour type 1 to behaviour 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 corresponding off-balance account → crediting the card contract's off-balance account. The corresponding 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 behaviour type 1 to behaviour 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 3-4" 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 corresponding off-balance account → crediting the card contract's off-balance account. The corresponding 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 3-4" 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".

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.** I.e. The date funds are first transferred from the standard account to the delinquency account is shown until the debt is fully repaid. The total number of days in the *Past Due Days* field is calculated from this date (i.e. 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 used the hardcoded PAST_DUE balance type (see Fig. 12.

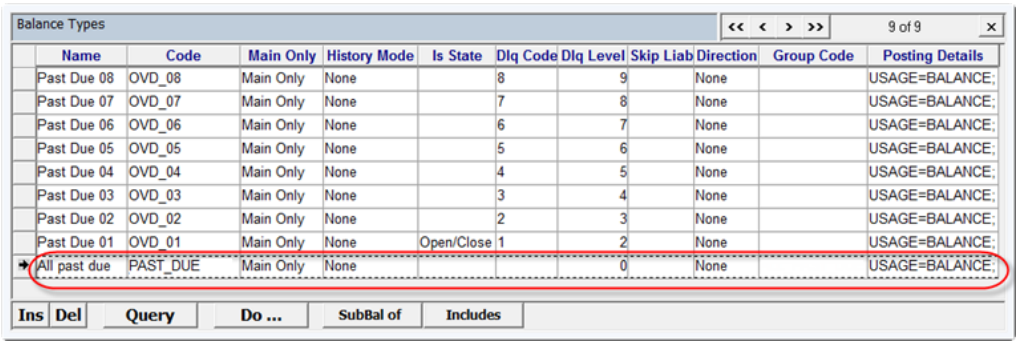


Fig. 12. Configuring a balance type to record the total amount of debt

  - Map all delinquency account templates with this balance type, see Fig. 13.

    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".

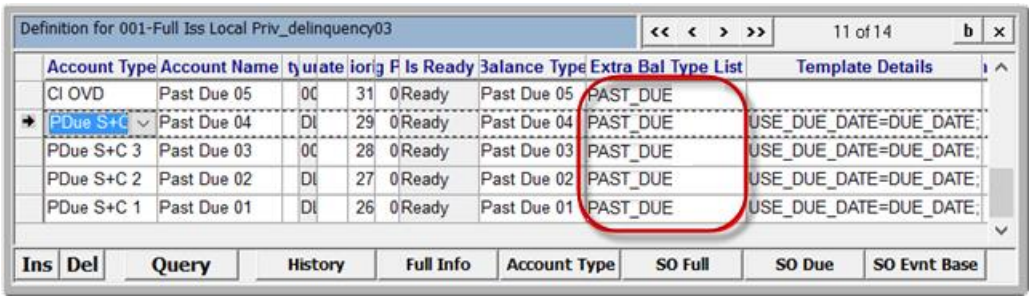


Fig. 13. Mapping account templates with the PAST_DUE balance type

  - 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. The 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 Accounting 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 can be used starting from version 03.44.30. Note that after these settings have been made, the new approach will be applied to new delinquency. Data for old delinquency (that arose before these settings were made) will not change (will not be recalculated).

Chapter 4. Working with Accounting Schemes

Working with Accounting Schemes consists of the following operations:

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

Copying Accounting Schemes

An Accounting Scheme can be created in the following ways:

- By clicking the [Ins] button to add an empty row and entering necessary data
- By copying an existing Scheme and changing its parameter values according to the requirements to the new Accounting 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 Accounting 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 "Accounting Scheme Approval").

Configuring Message Templates (Group Msg)

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

To set up a template of this message, use the [Group Msg] button in the "Details for <name of Accounting Scheme>" form (see Fig. 3 in section "Additional Parameters of an Accounting Scheme"). For more details on message template setup, see the "Configuration of Client Messages" document.

Configuring Events

Events for Accounting Schemes are configured in the "Events for <name of Accounting Scheme>" form, opened by clicking the [Events] button in the "Account Schemes" form (see Fig. 2 in section "Form "Account Schemes"").

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 details on setup of such Events, see section "Changing Contract Behaviour Types" in 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 details, see section "Event Types" in the document "Events".
- This form is used when configuring an Event intended to change the Accounting Scheme. For more information, see the section "Changing a Contract Accounting Scheme" of the document "Events".

Entering and Editing Tags

Forms "Details for <name of Accounting 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 Fig. 14) 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 Accounting Scheme>" and "Full Info for <name of account template>".

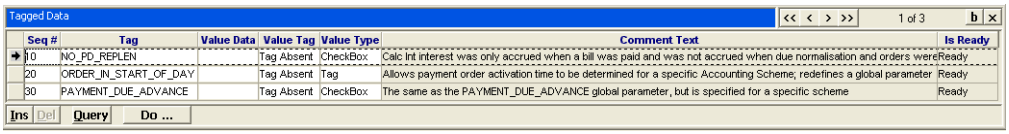


Fig. 14. Form "Tagged Data"

To add a tag, click the [Ins] button and fill in the fields of the 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.
- Field *Value Tag*:
  - 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 tag value
  - "Counter" – the value of the tag must be an integer (from "0" to "9")

  - "Tag" – the value of the tag must be either "Y" or "N"
  - "Money" – numeric tag 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 can 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 show 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 be opened.

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 Accounting Scheme>" form).
- "Save Tags" (in the "Full Info for <name of account template>" form).

Checking Accounting Schemes

Accounting Scheme parameters are checked for correctness in the form containing additional Accounting Scheme information (see Fig. 3 in section "Additional Parameters of an Accounting Scheme") by clicking the [Actions…] button and selecting the "Check" value from the context menu.

If errors are detected while checking Accounting 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.

Accounting Scheme Approval

After entering or modifying values of Accounting Scheme parameters, click the [Approve] button in the Accounting Scheme grid form (see Fig. 2 in section "Form "Account Schemes"" or Fig. 4 in section "Special Forms for Working with Accounting Schemes") to approve the changes.

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

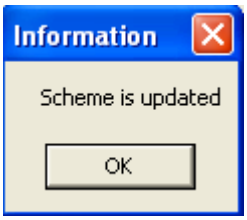


Fig. 15. Message informing that Accounting Scheme data has been updated successfully

When an Accounting 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 executed (for more details, see the WAY4™ Global Parameters Administrator Manual).

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

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

If in an Accounting Scheme an account template is deleted according to which accounts have already been created (records in the ACCOUNT) table, and activity has taken place on these accounts, this Accounting 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 Accounting Scheme.

Working with Additional Accounting Schemes (Included)

WAY4 provides functionality for attaching sets of additional accounts to main Accounting Schemes. To do so, follow the instructions below:

- Configure an additional Accounting Scheme with a set of required accounts in the form "Full → Configuration Setup → Products → Account Schemes" – for additional Accounting Schemes, specify the "Included" value in the *Used For* field of the Accounting Scheme (see the section "Form "Account Schemes"").

- Attach the configured Accounting Scheme to the main Scheme. To do so, follow the instructions below:
  - Open the form with additional parameters of the main Accounting Scheme to which the configured set of accounts should be attached (the value in the *Used For* field of this Accounting Scheme is either empty or the "Product" value is specified). See the section "Additional Parameters of an Accounting Scheme".
  - In the "Included for <name of main Accounting 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 attached Accounting Scheme.

  Only Schemes with the "Included" marker may be attached to Accounting Schemes.

When approving a main Accounting Scheme for which an attached 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 Accounting Scheme.
- If in the main Accounting Scheme there is an account template with the same type as in the attached Accounting 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 Accounting Scheme, this template's settings (i.e. 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*, and *Ageing Priority* fields, INT_GROUP tag value. I.e. the values of these parameters will not be synchronised with the attached template's parameters.

    When the INT_GROUP tag is specified in the main Accounting 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 attached, these changes are automatically inherited by the Main Accounting scheme when it is approved.

  Attachments are deleted in the "Included for <name of main Accounting Scheme>" form.

When deleting an attached Accounting 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 attachment, the link between the main Scheme and attached Scheme templates is deleted. Synchronisation of template changes is terminated and access to changing parameters of templates that were attached earlier is opened right in the main Accounting Scheme.

Chapter 5. Tags Used when Configuring Accounting Schemes and Account Templates

Name	**Value**	**Description**
Tags in the Special Parms *field* of an Accounting Scheme		
NO_PD_REPLEN		When this parameter is set, when transferring funds according to due normalisation rules to contract accounts with the value "Int By Credit" in the *Calc Int Mode* field, interest will not be additionally accrued. This rule is effective for accounts with the following normalisation types: "Payment Due", "Long Payment Due", "Sliding Due" *и* "Value Days Due".
ORDER_IN_START_OF_DAY	"Y" "N" "YP" "YN"	Allows the time payment orders are activate for a certain account scheme to be specified, redefines the global parameter of the same name. Values: • "Y" (Yes) – payment orders are processed during daily opening procedures; this is the default value. • "N" (No) – payment orders are processed during daily closing procedures. This value is only allowed if the bank uses the mode for separation of daily closing and opening. • "YP" – this value sets the procedure for processing payment orders by default, corresponding to the "Y" value. On the payment order level, exceptions can be set by setting the value of the ORDER_IN__START_OF_DAY tag to "N" in the Posting Details field. • "NP" – this value sets the procedure for processing payment orders by default, corresponding to the "*N"* value. *On the* payment order level, exceptions can be set by setting the value of the ORDER_IN__START_OF_DAY tag to "Y" in the Posting Details field.

Name	Value	Description
PAYMENT_DUE_ADVANCE	"Y" "N"	Redefines the global parameter of the same name for a certain Accounting Scheme. The parameter is used for daily processing of contracts and for due normalization types "Value Date Due", "Payment Due", "Long Payment Due", "Sliding"and "Sliding + Clear" determines the macrotransaction posting date and date of corresponding entries to accounts (Local Date). For more information, see the section "PAYMENT_DUE_ADVANCE" of the document "Global Parameters".
BILLING_EVNT	"E" "S"	For Events with the "Billing" value in the *Duration Type* field allows the Event to be closed at the end of the billing cycle: • "E" – when closing the last banking day of a billing cycle. • "S" (default value) – when opening the first day of a new billing cycle. The tag can be redefined in the Event type.
MOVE_RESERVE	An arbitrary value. For convenience, we specify the "MOVE" value in documentation.	When a balance is changed on an account for reserving, allows a direct entry to be made between accounts of the created reserve.
TO_WRK_DAY		When a billing cycle is shifted (if the end of the billing cycle does not correspond with the end of the month), if the first day of the new billing cycle is a weekend/holiday, this parameter allows the first day of the new billing cycle to be shifted to the next working day. Example1. The last day of a contract's billing cycle is the 10th. In June, this date (10.07.15) falls on a Friday. I.e. the first day of the new billing cycle falls on a Saturday (11.07.15). Billing cycle dates will be calculated as follows: Billing cycle start date (last_billing_date) – 11.06.2015. Billing cycle end date (next_billing_date) – 12.07.2015. Therefore, the start date of the next billing cycle will be Monday, 13.07.2015. Example2. The last day of a contract's billing cycle is the 23rd. In

Name	Value	Description
		August, the end of the billing cycle (23.08.2015) falls on a Sunday. Billing cycle dates will be calculated as follows: BIlling cycle start date (last_billing_date) – 24-07.2015. BIlling cycle end date (next_billing_date) – 23.08.2015. The start date of the next billing cycle will be Monday 24.08.2015.
MIN_BILLING	<minimum billing cycle length in days> "C"	This parameter allows the minimum billing cycle to be specified in days. If for a new contract the length of a billing cycle is less than the value of this parameter, the end date of the next billing cycle will be used as the end date of the first billing cycle. The MIN_BILLING=C; tag specifies that there may be only one Billing Date during a calendar month. If this condition is not met, a billing cycle's end date is shifted to the end date of the next billing cycle.
USE_DATE_OPEN		The USE_DATE_OPEN parameter (tag) affects how the start date of the first billing cycle is determined if the length of a billing cycle is set in months. The USE_DATE_OPEN; tag is used when interest must be charged for a contract's first billing cycle, starting from the contract's opening date, and not for the entire corresponding billing cycle (if billing cycle length is set in months). If the USE_DATE_OPEN tag is set, the contract's opening date will be the start date of the first billing cycle, on the condition that the contract's opening date is equal to the current banking date. If the contract's opening date is earlier than the current banking date (this situation may occur when Billing Date is recalculated, see the document "Contract Functional Dates"), the current banking date will be used as the start date of the first billing cycle. If the USE_DATE_OPEN tag is not set, the start date of the first billing cycle is calculated as follows: The end of the first biling cycle is calculated The start date of the first billing cycle is determined

Name	Value	Description
		according to the formula: End date of the first cycle-1 month+1 day I.e. if a contract is opened 12/12/2016, the following dates are calculated for it: The end date of the first billing cycle is 31/12/2016 The start date of the first billing systel is 01/12/2016 (31/12/2016 - 1 month + 1 day If the billing cycle is shifted relative to the start of the month and the contract is opened 12/12/2016: The end date of the first billing cycle, for example, 11/01/2017 The start date of the first billing cycle 12/12/2016 (11/01/2017 - 1 month + 1 day). I.e. in this case, the billing cycle's start date will be the contract's opening date.
OVL_VARIANCE	<number of percentage points> <tariff type code>	These tags make it possible to not open Events with the predefined codes OVL_BILLING and OVL_DAILY if the amount of overdue (amount on the unauthorised overdraft account (OVL)) is less than the amount specified in the tag OVL_VARIANCE_AMNT or less than a percentage of the credit limit amount specified with the OVL_VARIANCE tag. If both threshold values are set (OVL_VARIANCE_AMNT and OVL_VARIANCE), Events will be opened if the amount of overdue exceeds the lesser value specified. A "Threshold" tariff type code can be specified as the tag value.
OVL_VARIANCE_AMNT	<amount> <tariff type code>	
PAYOFF_VARIANCE	<number of percentage points> <tariff type code>	Makes it possible at the end of a billing cycle to generate an Event with the predefined code FULL_PAYMENT if the amount of the due payment is less than specified in PAYOFF_VARIANCE_AMOUNT or the percentage of the total amount of debt set by the PAYOFF_VARIANCE tag. A "Threshold" tariff type code can be specified as the tag value.
PAYOFF_VARIANCE_AMOUNT	<сумма> <код типа тарифа>	

Name	Value	Description
EOM_INTEREST	"ALL" "IF_CREDIT" "BY_ACC"	Allows automatic accrual of interest on the last day of a billing cycle (when *the* end of a billing cycle doesn't correspond with the end of the month): • "ALL" – on all accounts of the accounting scheme (for which interest accrual is configured). • "IF_CREDIT" – on all accounts of the accounting scheme for which interest accrual is configured and the parameter "Calc Int Mode" is set. • "BY_ACC" – *on* accounts that have the parameter EOM_INTEREST in Template Details field of the account template.
SPLIT_MONTHLY		When this tag is specified, closing turnover on accounts at the end of a calendar month is forced for contract accounts of this Accounting Scheme This tag should be used carefully, since it increases the daily closing procedure by about 1.5 times (once a month) and increases the amount of the ITEM table.
NO_COPY		When this tag is set, the link is broken between this Accounting Scheme and *the Scheme* specified in the Parent Scheme. After this tag is set and changes are approved, the parameters of the parent Scheme will not be copied to this Accounting Scheme.
INT_IN_CYCLE		This tag redefines the global parameter INTEREST_IN_CYCLE and the parameter "Interest in Cycle" of the financial institution.
DUE_MODE	"FIXED_DAY"	The tag is used to calculate the day of an Event's opening for Events with the *predefined* codes FULL_PAYMENT and LATE_PAYMENT. The tag with the "FIXED_DAY" value allows this Event *to* be opened on a certain day of the month (<Due Date field value> - 1).
MIN_BILLING_UNIT	"M"	The parameter is used if the billing cycle length cannot be less than a month. When the parameter MIN_BILLING_UNIT=M; parameter is set, if the length of the first billing cycle for a new contract is less than a

Name	Value	Description
		month, the end date of the next billing cycle will be used as the end date of the first billing cycle.
MTP_ROUND	"n" "-m" ">k" "<l" "=p"	The MTP_ROUND=<value> tag makes it possible to round the calculated minimum payment to a specified sign. Possible tag values: • "n" – round to 10-n ("0" to the basic unit of currency, dollars, pounds), "1" to a fractional currency unit (cents, pence), "2" - to units of a fractional currency; in both cases it is assumed that the fractional currency portion of the amount is 2 decimal places); "1" to tenths of the basic currency unit, etc.) • "-m" – rounding to 10-m ("-1" to tenths of the basic currency unit, "-2" to hundredths of the basic currency unit) • ">k" – rounding up to k fractional currency units • "<l" – rounding down to l fractional currency units • "=p" – rounding according to a standard algorithm to p fractional currency units
GL_TRF		The tag is specified automatically if the Accounting Scheme has at least one template with the *GL* Tariff field filled in (with the specified tariff type with the "GL Numeration" role). The tag cannot be edited manually.
EOM_INT_MODE	"Y" "P"	The EOM_INT_MODE tag can be used when accruing interest at the end of a calendar month if rules for recording interest accrued at the end of the month differ from rules for accruing interest at the end of a billing cycle. The EOM_INT_MODE tag makes it possible to use different rules for posting these entries to GL accounts. The EOM_INT_MODE tag redefines the global parameter INTEREST_IN_CYCLE (or the INT_IN_CYCLE tag set in an Accounting Scheme) with regard to specifying the GL Date of macrotransactions for accruing interest at the end of the month (if the end of the billing period corresponds with the end of the month. Parameter values: "Y" – date of recording entries in the General Ledger

Name	Value	Description
		(Local Date) is the last calendar day of the month "P" – date of recording entries in the General Ledger (Local Date) is the last working day of the month. For more information, see the section "EOM_INT_MODE" of the document "Interest Accrual".
LP_VARIANCE	<percentage number> <tariff type code>	The tags are used to set conditions for charging interest for late payment. These tags make it possible to not open an Event with the predefined code LATE_PAYMENT (i.e. waive interest) if the amount of unpaid debt is less than the amount specified using the LP_VARIANCE_AMOUNT tag or less than the percentage of the amount of debt specified using the LP_VARIANCE tag. If both threshold values are set (LP_VARIANCE_AMOUNT and LP_VARIANCE tags), an Event will not be opened if the overdue amount exceeds the lesser of the specified values. The code of the"Threshold" tariff type used to set the corresponding threshold can be set as the tag value.
LP_VARIANCE_AMOUNT	<amount> <tariff type code>	
LP_BALANCE	<balance type code>	The tag is used to set conditions for charging interest for late payment (i.e. to set conditions for triggering an Event with the LATE_PAYMENT code). The LP_BALANCE tag makes it possible to set the code of the balance type for analysing overdue debt. (By default, the TOTAL_DUE balance code is used). Used together with the LP_VARIANCE and LP_VARIANCE_AMOUNT tags.
FP_BALANCE	<comma-delimited list of balance type codes>	The tag FP_BALANCE=<comma-delimited list of balance type codes>; is used to set a grace (interest-free) period for certain balance types (i.e. set up triggering an Event with the FULL_PAYMENT code when paying a certain balance type). For example: FP_BALANCE=STMT_RETAIL,STMT_CASH; (separation into "Cash" and "Retail"). The tag is used together with the FP_BALANCE=<balance type code>; tag in the Accounting Scheme, and the FP_<balance type

Name	Value	Description
		code>_VARIANCE_AMOUNT, FP_<balance type code>_VARIANCE tags.
FP_<balance type code>_VARIANCE	<percentage> <tariff type code>	The tags are used to set a grace (interest-fee) period when payment is made on time. These tags make it possible to open an Event with the code FP_<balance type code>, and waive interest for the respective accounts if the amount of unpaid debt is less than the amount specified with the tag FP_<balance type code>_VARIANCE_AMOUNT, or less than the percentage of the debt amount specified with the FP_<balance type code>_VARIANCE tag. If both threshold values are set, Events will open if the overdue amount exceeds the smaller of the specified values. Used together with FP_BALANCE tags in the Accounting Scheme and account template. The code of the "Threshold" tariff type used to set the corresponding threshold can be set as the tag value.
FP_<balance type code>_VARIANCE_AMOUNT	<amount> <tariff type code>	
EOC_TO_WRK_DAY		If a billing cycle is shifted (if the end of the billing cycle does not correspond with the end of the month) and the last day of the closing billing cycle falls on a weekend/holiday, this parameter makes it possible to shift the last day of the billing cycle to the next working day.
CALENDAR_TYPE	<name of business calendar type>	Allows separate rules for calculating non-working/working days for posting due normalisation macrotransactions to be set for a scheme's account templates (when the SHIFT_TO_WRK_DAY parameter is set globally or in the account and the DUE_MODE=WRK_DAY tag is set in the account). To do so, a separate business calendar differing from the financial insitution's business calendar is set as the tag value.

Name	Value	Description
USE_MONTH_WEIGHT	"B"	USE_MONTH_WEIGHT=B; redefines the global parameter of the same name. I.e. the algorithm for accruing interest corresponding to this value can be set in an Accounting Scheme. For more information, see the section "USE_MONTH_WEIGHT" of the document "WAY4 Global Parameters".
INST_SCHEDULED_ER_BAL	<balance type code>	The tag is used in the WAY4 Instalments module to set up scheduled early repayment. The tag redefines the value of the same global parameter. For more information, see the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
INST_AUTO_ER_BAL	<balance type code>	The tag is used in the WAY4 Instalments module to set up automatic early repayment. The tag redefines the value of the same global parameter. For more information, see the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
CALC_PD_DATE		The tag is used to show the date delinquency arose and the total number of past due days in Customer Service Workbench. See the section "Configuring Display of Past Due Date and Past Due Days in Customer Service Workbench" of the document "WAY4 Accounting Schemes". 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).

Name	Value	Description
FIRST_BILLING	<number of days> ANY <number of months>	The value of the FIRST_BILLING tag redefines the value of the MIN_BILLING tag when calculating the end date of the first billing cycle. The FIRST_BILLING=ANY; tag makes it possible to set any length for the first billing cycle. The FIRST_BILLING=<number of days>; tag makes it possible to set the minimum length of the first billing cycle, in days. FIRST_BILLING=<number of months>;FIRST_BILLING_UNIT=M; tags make it possible to set the minimum length of the first billing cycle in months.
FIRST_BILLING_UNIT	M	FIRST_BILLING=<number of months>;FIRST_BILLING_UNIT=M; tags make it possible to set the minimum length of the first billing cycle in months.
MAX_BILLING	<maximum billing cycle length in days>	The tag sets the maximum length of a billing cycle. The tag is only checked when recalculating the current billing cycle's end date. If the length of the current billing cycle must be increase when changing the billing cycle's end date, the MAX_BILLING tag is checked. A check is made that the length of the billing cycle does not exceed the length in days specified as the value of the MAX_BILLING tag. If it does, the current billing cycle remains unchanged and the new Billing Date is only applied in the next billing cycle.
MAX_BILLING_UNIT	M	The tag is used if the length of a billing cycle may not be more than a month. The tag is used together with the MAX_BILLING tag. The tags are only checked when recalculating the current billing cycle's end date. When MAX_BILLING_UNIT=M; is set if the length of the first billing cycle for a new contract is more than a month, the date will bot be recalculated and the current billing cycle will remain unchanged.

Name	Value	Description
INTEREST_CONTRACT	FROM_DECISION:<decision code> INTEREST_CONTRACT=FROM_TARIFF:<tariff type code>	Makes it possible use decisions or tariffs to redefine an interest contract. The tag value can be specified with the ":STRICT" postfix. For example, , INTEREST_CONTRACT=FROM_DECISION:REV_CONTR_ISS:STRICT;. In this case the bank contract number specified is used as it is set in classifier/tariff parameters. By default (if the ":STRICT" postfix is not set) this contract number is used as a template. I.e. the prefix of the corresponding financial institution (the financial institution for the contract with which the operation is being made (Target or Source contract)) is automatically added to it. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes".
Tags in the *Spec Parms* field of an Account Template		
CUST_GL_MODE		Used to call the custom procedure CUST_GL_NUMBER. For more information, contact WAY4 vendor representatives.
XF	<Account type code 1>:<Account type code 2>	When several balance accounts are used for correspondence with the off-balance subsystem – if the bank contract balance account depends on the off-balance account to which funds must be transferred (for example, if various revenue accounts must *be used),* the OffBalance XF Acc field of the account template is not used. In this case, the XF tag is used, which allows account pairs to be set that participate in entries during interaction of the balance and off-balance accounting subsystems. The format for specifying the parameter value is "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

Name	Value	Description
		<Bal_Default> – default bank contract balance account type code
CONTRA_ORDER	<Order code+> <Order code ->	Allows a standing payment order with a set code to be called when debiting and crediting this account. When crediting, the order <Order code+> will be activated, when debiting – <Order code->
WAIVED_PD_TO	<Code of the account type to which the balance is transferred>	If the amount in the account is less than that specified with the WAIVED_PD_AMOUNT parameter, the balance from this account is transferred to the account with the code specified in WAIVED_PD_TO. An account with this code is first searched for in the same contract, and if it is not found, a search is made *according* to the accounts of the contract specified in the Interest Contract field of the account template.
WAIVED_PD_AMOUNT	<Account balance amount>	If the amount in the account is less than that specified as the value of the WAIVED_PD_AMOUNT parameter, the balance from this account is transferred to the account with the code specified in WAIVED_PD_TO (see above). The amount is specified in the account currency.
CREATE	"ALL" "MAIN"	The CREATE parameter is used to configure the procedure for creating contract accounts (in addition to the account template's GL Properties parameter group): When the value is "MAIN", accounts of the main contract are created on the first approval of the respective contract (or when an Account Scheme is approved if it is approved after approval of the contract). When the value is "ALL", accounts of the main contract and of subordinate contracts are created on the first approval of the respective contract (or when an Account Scheme is approved if it is approved after approval of the contract). The CREATE parameter can only be used when the value of the global parameter CREATE_ALL_ACCOUNTS is "N".

Name	Value	Description
WAIVE_INT_EVNT	<Event type code>	Makes it possible to not accrue interest on an account when an Event with the specified code is active.
ROUND	"n" "-m" ">k" "<l" "=p"	ROUND =<value> – allows interest accrued on this *account* to *be* rounded to the specified sign. Possible values of the tag: • "n" – rounding to 10-n ("0" –to a basic unit of currency, dollars, pounds), "1" – *to a* fractional currency, cents, pence), "2" – to units of fractional currency; in both cases it is assumed that the fractional currency portion *of the* amount is 2 decimal places. I.e. the amount of 20.78 USD will *be* rounded to 20.80 (to tenths) when the value is "1" (ROUND=1;) and when the value is "2" (ROUND=2;), it will remain the same. • "-m" – rounding to 10-m ("-1" – to tenths of the basic unit of currency, "-2" to hundredths of the basic unit of currency). For example "ROUND=-2"; makes it possible to round 244.64 USD to 200 USD. • ">k" – rounding up to k fractional currency units. For example, when ROUND=>1000; the amount of 244.64 USD is rounded to 250.00 USD). When ROUND=>100;, the amount of 244.64 USD is rounded to 245.00 USD). Note. ROUND=>200 setup is used for rounding to even units of the main currency (for example, to 2/6/4/8 USD depending on the rounded amount). For example, when a fee rate is 3% and the transaction amount is 100 USD, the initially calculated fee amount of 3 USD will be rounded to 4 USD. • "<l" – rounding down to l fractional currency units. For example, when ROUND=<1000; the amount of 244.64 USD is rounded to 240.00 USD). When ROUND=<100;the amount of 244.64 USD is rounded to 244.00 USD). • "=p" – rounding according to a standard algorithm to fractional currency units. It is recommended to use this value to set up rounding to "p" units, hundredths of the main currency (for units, hundredths of dollars, pounds).

Name	Value	Description
		I.e. it is recommended to use "100", "1000", etc. When ROUND==1000;, the amount of 125.65 is rounded to 130. When ROUND==100;, the amount of 125.65 is rounded to 126. When it is necessary to round to tenths, units of fractional currency (i.e. rounding the part of the amount "after the decimal"), use the ROUND=n; tag, see above.
WAIVE_INT_ROUNDING	"Y" "N"	Redefines the value of the global parameter of the same name. Used when accruing interest on instalment loans if interest is accrued several times during one billing cycle (when accruing interest rounding errors may occur connected with rounding to the minimum fractional currency unit; for example, for dollars – to two places after the comma); • The "N" value allows consideration of the amount of a rounding error occurring during interest accrual in the next accrual of interest. When interest is next accrued in the same billing cycle, the amount of the rounding error not considered earlier is added to the calculated amount. • When the value of this global parameter is "Y", the rounding error amount is not considered.
ROUND_FEE	"n" "-m" ">k" "<l" "=p"	Makes it possible to round the calculated Interest Fee on this account to the specified decimal place. See the description of values in the ROUND parameter.
INT_FEE_SRC_ACC	<account type code>	Allows interest to be accrued at the rate set in *the Interest* Fee Rate field, to a separate account, by a separate entry (allows separation of interest accrual by Interest Rate and Interest Fee Rate). *To do so*: • In the *Fee Rate Mode* field, select the "Direct" value. • The code of the Accounting Scheme account on which interest is accrued is specified as the value of the INT_FEE_SRC_ACC tag. • In the *Interest Fee Account* field, the bank account

Name	Value	Description
		participating *in the* interest accrual entry is specified (the contract is set in the Interest Contract field). If the Interest Fee Account parameter is set in the template and the account code is set in the INT_FEE_SRC_ACC tag, a search is made for an account template in this Accounting Scheme with the specified code and currency of the original account. The currency of the template being searched for can be redefined using the tag INT_FEE_SRC_ACC_CURR. If the appropriate template is not found in the Accounting Scheme, a warning message will be displayed. If the template is not found, the template specified in the Interest Template field will be used.
SKIP_DUE_EVNT	<Event type code>	The SKIP_DUE_EVNT parameter makes it possible to disable due normalization on an account if an Event with the specified code is open (makes it possible to "freeze" the amount on an account). The parameter can be used for "End Cycle Due" and "Quarter" due normalization types (if the amount is not yet due, i.e. doesn't work for Payment Due accounts).
CUST_DUE_AMNT	<Event type code>	Makes it possible to define the amount of due normalisation using the custom procedure CUST_DUE_AMOUNT
FROM_NEXT_BILLING		This tag remains for backward compatibility. Use of the tag is not recommended. For account templates with the "Payment Due" and "Long Payment Due" type, makes it possible to begin counting the due period from the next cycle's start date, and not from the *start* date of the current cycle (for example, when the Due Type parameter value is "Payment Due" and the Due Period parameter value is "-1", one day is subtracted from the start date of the next billing cycle; i.e. in this setup, normalization is executed on the last day of a billing cycle). In this setup, if funds are transferred sequentially between several accounts within one normalization process, the amounts transferred to the

Name	Value	Description
		account are not totally with the amount on this account intended for transfer to another account. It is not recommended to use this tag in accounts with a Due Type parameter value other than "Payment Due"/"Long Payment Due" since this leads to incorrect calculation of the normalisation date compared with due normalisation settings in the account template.
INT_RATE_MODE	"MIN" "MAX" "AVG"	This parameter makes it possible to set rules for selecting the interest rate depending on the account balance using tariffs. Possible parameter values: • "MIN" – the interest rate is selected depending on the minimum value of the account balance for the billing cycle. • "MAX" – the interest rate is selected depending on the maximum value of the account balance for the billing cycle. • "AVG" – the interest rate is selected depending on the average value of the account balance for the billing cycle. To implement this functionality, use the "Service Limit" tariff role (for more information, see the document "Advanced Tariff Management). The Advanced Tariff Management module is not included in the WAY4 basic configuration and is delivered according to an additional agreement with the WAY4™ vendor.
SHIFT_TO_WRK_DAY	"Y" "N"	Redefines the value of the global parameter SHIFT_TO_WRK_DAY. If it is set to Y, for accounts with due normalization types "Payment Due" and "Sliding Due", the due date will be shifted to the nearest business day. If SHIFT_TO_WRK_DAY is used (unlike the global parameter DUE_TO_WRK_DAY), the date of activity in the contract account (Statement Entry) is corrected and the corrected value is shown in the statement.

Name	Value	Description
DUE_MODE	"WRK_DAY" "MONTH" "BILLING"	Allows a Due period to be set in working days, months and billing cycles. If the tag is not specified, the due period is measured in calendar days. Works for all due types, except for "Value Date Due", "Fixed Day Due" and "End Cycle Due".
OPPOSITE_RSRV_RATE		When reserving a balance on this account with the parameter OPPOSITE_RSRV_RATE set, the balance will be taken with the opposite sign. This setting is used, for example, for accounts with unused credit lines, so that when reserving accounts with *a positive* balance, a negative value is not set in the Risk Rate field of the reserving scheme.
EOM_INTEREST	"IF_CREDIT" "ALL" "<arbitrary value differing from those listed above>"	This parameter allows interest accrual on accounts at the end of the month (when the end of a billing cycle doesn't correspond with the end of the month): • "ALL" – for all Scheme accounts with the set interest accrual parameters. • "IF_CREDIT" – for all Scheme accounts for which interest accruals configured and the "Calc When Credit" parameter is set. • If EOM_INTEREST=<arbitrary value differing from other values of this parameter>; is set in the Accounting Scheme and the parameter EOM_INTEREST; is set in the account template, interest on this account will be accrued on the last day of the month.
CALENDAR_TYPE	<name of business calendar type>	Allows to create for this account rules for calculating weekend/working days for activation of Due normalization, when the parameters SHIFT_TO_WRK_DAY and DUE_MODE=WRK_DAY are set on the account. To do so, a separate business calendar differing from the financial institution's business calendar is set as the parameter value.
CALC_WHEN_CREDIT_EVENT	<Event type code>	Allows to enable forced posting of accrued interest on a loan when repayment is made (when the "Calc Int Mode"

Name	Value	Description
		parameter is set) if an Event with this code is open for this contract.
INVOICE_CODE		This tag is used when working with the WAY4 Instalments module to search for an instalment scheme. The tag value is the *Invoice Code* field value of the corresponding instalment scheme). See the document "Instalment Loans in WAY4". The module is delivered according to a separate agreement with the WAY4 vendor.
SHIFT_DATE_FROM_TO_WRK_DAY	"Y" "N"	The tag redefines the global parameter of the same name, SHIFT_DATE_FROM_TO_WRK_DAY, that makes it possible when defining a normalization period to shift the start date of this period to a working day, if the original date falls on a non-working day. • If the global parameter's value is "Y", this mode can be redefined on the account template level using the SHIFT_DATE_FROM_TO_WRK_DAY tag with the "N" value (disables date shifting in a certain template). • If the global parameter's value is "N", settings with the SHIFT_DATE_FROM_TO_WRK_DAY tag are ignored in the account template. • If the global parameter's value is "C", the start of the normalization period will be calculated based on the account template. If the SHIFT_DATE_FROM_TO_WRK_DAY ("Y", "N") tag is set in the template, these settings are used when calculating the period. If the tag is not set, dates are not shifted and default system behaviour is supported.
ALLOW_ZERO_DUE	"Y"	The ALLOW_ZERO_DUE tag is used to enable due normalization for a null amount for a certain account template. The tag ALLOW_ZERO_DUE=Y; is set in the template of the target account. The *Due Type* parameter of this template must have the value "Value Due Date". In addition, the ALLOW_ZERO_ENTRIES; tag must be set in the corresponding Service and transaction types for payment orders (for "Debit" and "Credit" orders).

Name	Value	Description
		In this case (if there is even one entry for a null amount for the account), a macrotransaction is generated for a null amount. The macrotransaction is used to link account entries with a total null amount with standing payment orders (invoices); i.e. to activate a payment order for a null amount (issue an invoice for a null amount). For example, when a transaction is reversed, if there are two documents for opposite amounts.
USE_DUE_DATE	<functional date code>	The tag USE_DUE_DATE=<functional date code>; (for example, USE_DUE_DATE=LP_DATE;) sets the date that will be used to process due normalisation for account templates with the "*Contract Due*" value of the Due Type parameter. By default, to process due normalisation, the date "Delinquency Date" (DLQ_DATE) or "Due Date" (DUE_DATE) is used if the DLQ_DATE date is not set.
INT_TOLERANCE	<percentage number> <tariff type code>	The tags make it possible to forgive unpaid loan interest at the end of a billing cycle, if this amount is less than the value specified with the INT_TOLERANCE_AMOUNT tag or less than the percentage of the contract's credit limit set using the INT_TOLERANCE tag (a credit limit is determined by the balance type with the predefined code FIN_LIMIT). The INT_TOLERANCE_AMOUNT tag is used to set an amount in the account template currency. If both tags are set, the interest amount will be compared with the lower value. The tags can be used if interest is accrued once, at the end of a billing cycle. These limiters can be set up using a tariff with the "Threshold" role: • without tags. In this case, the tariff is created with the INT_TOLERANCE code. • with tags. In this case, the tag value is specified as the code of the "Threshold" tariff type used to set the corresponding threshold.
INT_TOLERANCE_AMOUNT	<amount> <tariff type code>	
INT_FEE_TOLERANCE	<percentage number> <tariff type code>	The tags make it possible at the end of a billing cycle to forgive an interest fee (Interest Fee Rate). The fee is

Name	Value	Description
INT_FEE_TOLERANCE_AMOUNT	<amount> <tariff type code>	forgiven if the fee amount is less than the value specified with the tag INT_FEE_TOLERANCE_AMOUNT or less than the percentage of the contract credit limit set using the INT_FEE_TOLERANCE tag (a credit limit is determined by the balance type with the predefined code FIN_LIMIT). Tags are set in the template of the account for which this fee is charged. These limiters can be set up using a tariff with the "Threshold" role: • without tags. In this case, the tariff is created with the INT_FEE_TOLERANCE code. • with tags. In this case, the tag value is specified as the code of the "Threshold" tariff type used to set the corresponding threshold.
MIN_DUE	<tariff type code>	This tag sets the code for the tariff type with the "Threshold" role used to define the minimum amount of debt for transferring funds to the next overdue account. For more information, see the document "WAY4™ Advanced Tariff Management". The amount set with the tariff is compared *with the* account balance (either in the account of due funds or in the oldest delinquency account). If the account balance is less than the amount set in the tariff, funds are not moved to the next delinquency account. A transfer will not be made only if there are non-empty balances in accounts with a higher delinquency level than the current account. It is not recommended to use this tag in accounts with the "Sliding", "Sliding + Clear", "Value Day Due" values of the *Due Type* parameter since if when the due date arrives funds are not transferred to a delinquency account (to the next delinquency account) and stay in this account, a new instalment payment date will not be calculated. For accounts with the "Sliding + Clear" value of the *Due Type* parameter, the MIN_DUE tag can be used except when a small balance remained in the account (less than the minimum amount set in the MIN_DUE tariff) after which

Name	Value	Description
		account activity stopped. If new debt does not arise for the contract and payment is not made, the MIN_DUE tariff for the contract is disabled, the balance is no longer checked for compliance with the MIN_DUE limit and the balance does not "age" (doesn't go to other delinquency accounts). In this case, the balance does not "age" until the next activity in the account.
WAIVED_PD	<tariff type code>	This tag sets the code for the tariff type with the "Threshold" role used to define the minimum amount of debt for returning funds to the previous account for recording funds to be paid. For more information, see the document "WAY4™ Advanced Tariff Management". This tag is used when configuring an account with the "End Cycle Due" or "Sliding + Clear" value of the *Due Type* parameter.
CLEAR_USAGE_GROUP	<value of the limiter's Group Code field>	This tag makes it possible to reset limiter counters of contracts with a specific value in the GROUP_CODE field (update records in the USAGE_HISTORY table). Used when setting up a limiter (limit) to issue cash by credit card (see the section "Limiting Cash Withdrawal with a Credit Card (Configuring a Credit Limit for Cash Withdrawal Transactions)" of the document "Usage Limiters".
FP_BALANCE	<code of a balance type specified in the Accounting Scheme>	The tag is used to set a grace (interest-free) period for certain balance types (i.e. set up triggering an Event with the FULL_PAYMENT code when paying a certain balance type). The tag is used together with the FP_BALANCE tag in the Accounting Scheme. In account templates where *Calc* Int Mode = Waive after Full Payment (for which interest must be waived), specify the tag FP_BALANCE=<the code of a balance type specified in the Accounting Scheme>; according to the account's balance type. For example, in CL Loan Cash, specify FP_BALANCE=STMT_CASH;

Name	Value	Description
INT_FEE_SRC_ACC_CURR	<currency code>	The tag makes it possible to redefine the currency when searching for an account set with the INT_FEE_SRC_ACC tag. By default, an account is searched for in the original account's currency.
CHECK_BAL	"N"	The tag makes it possible to not check available funds when making a transaction, when the following conditions are observed: 1. The CHECK_BAL=N; tag is set in the Service 2. The CHECK_BAL=N; tag is set in the template of the account used for the transaction. *3*. The *Is Am Av* field value of the account used for the transaction is "No"
USE_MONTH_WEIGHT	"B"	USE_MONTH_WEIGHT=B; redefines the global parameter of the same name. I.e. the algorithm for accruing interest corresponding to this value can be set in an account template. For more information, see the section "USE_MONTH_WEIGHT" of the document "WAY4 Global Parameters".
STORNO_TYPE	"Y" "N" "B" "D"	The tag refines the same global parameter. For more information, see the section "STORNO_TYPE" of the document "WAY4™ Global Parameters".
XF_ROUND;		The XF_ROUND; tag together with the PRECISION tag defines rules for rounding and transferring the amount accumulated in a “High Precision” cumulative account to a standard account. For more information, see the section “High Precision Accounting” of the document “WAY4 Accounting Schemes”. !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 normalisation *(with* "Account Due" in the *Date Event* field).
MIN_TOTAL_DUE	<tariff type code>	This tag sets the code of the "Threshold" tariff type used to set the minimum total amount of debt for a contract to

Name	Value	Description
		transfer funds from this account to the next delinquency account. The amount set using the tariff is compared with the TOTAL_DUE balance type amount. If the TOTAL_DUE balance amount is less than the amount set by the tariff, funds are not transferred to the next delinquency account. If the MIN_DUE tag is used in the account template, the MIN_TOTAL_DUE tag is ignored.
INT_RESET_DATE	<date code>	To recalculate an interest rate only on a certain contract functional date (for example, on Due Date), specify the INT_RESET_DATE=<date code. Tag (for example INT_RESET_DATE=DUE_DATE;) in the account template. In this case, when a rate is changed with a tariff, the rate will be applied from the functional date (Due Date) following the date the tariff was changed. The tag only works if a tariff with the "Interest" role is used in the account template. See the section "Tariffs with the "Interest" Role" of the document "WAY4 Advanced Tariff Management". !The INT_RESET_DATE=DUE_DATE setting cannot be used in current accounts (accounts for which transactions are recorded, for example Current Loan, etc.). This setting may only be used in accounts to which money is transferred *as* the result of normalization (due normalization at the end of a billing cycle or on a DUE_DATE). !If the "From Start Billing" value is set in the *Apply Mode* field of the tariff, the INT_RESET_DATE=DUE_DATE; tag in the account template redefines the tariff's setting (i.e. the rate is revised on the specified functional date (Due Date)). !If transactions are corrected using the Reverse Management module, when the INT_RESET_DATE tag is set and the interest rate is changed on a certain date in the past, calculation is performed as follows: • The interval between the date on which the tariff was changed and the functional date when the rate became effective is recalculated according to the old rate.

Name	Value	Description
		• Starting from the functional date following the date on which the tarriff was changed, recalculation is performed according to the new rate.
INTEREST_CONTRACT	FROM_DECISION:<decision code> INTEREST_CONTRACT=FROM_TARIFF:<tariff type code>	Makes it possible use decisions or tariffs to redefine an interest contract. Works for all Accounting Scheme templates. May be redefined in a separate account template. The tag value can be specified with the ":STRICT" postfix. For example, , INTEREST_CONTRACT=FROM_DECISION:REV_CONTR_ISS:STRICT;. In this case the bank contract number specified is used as it is set in classifier/tariff parameters. By default (if the ":STRICT" postfix is not set) this contract number is used as a template. I.e. the prefix of the corresponding financial institution (the financial institution for the contract with which the operation is being made (Target or Source contract)) is automatically added to it. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes".

Chapter 6. High Precision Accounting

"High Precision" accounts are WAY4 technical internal accounts that do not participate in standard accounting. These accounts are used to record fund activity 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).

General Scheme of Operation

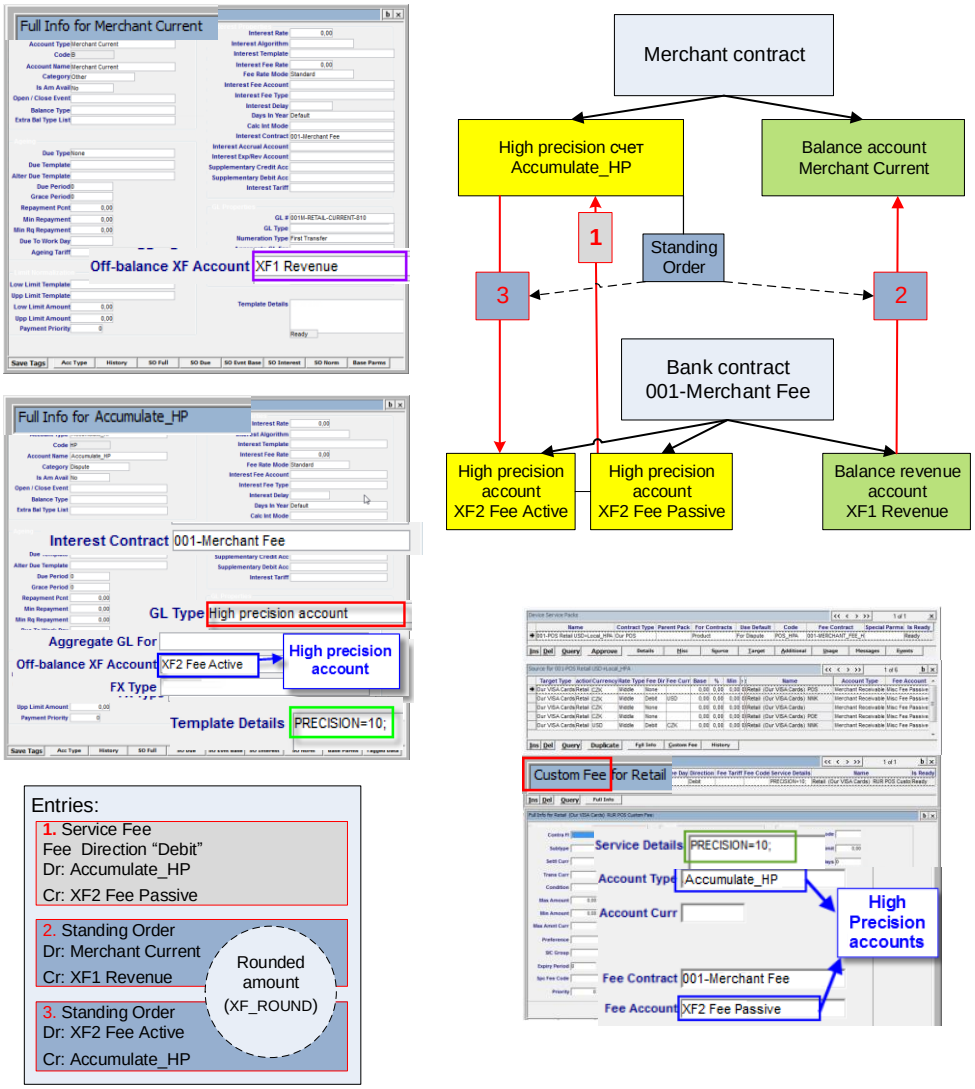


Fig. 16. Example of general account setup scheme for high precision recording of fees (without settings in the GL_TYPE table).

Service Settings

Additional settings for the Service of a high precision custom fee (see Fig. 17):

- 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 Fig. 17).
- 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 "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".

Fig. 17 shows an example of setup for charging a custom fee (with high precision) from a merchant contract.

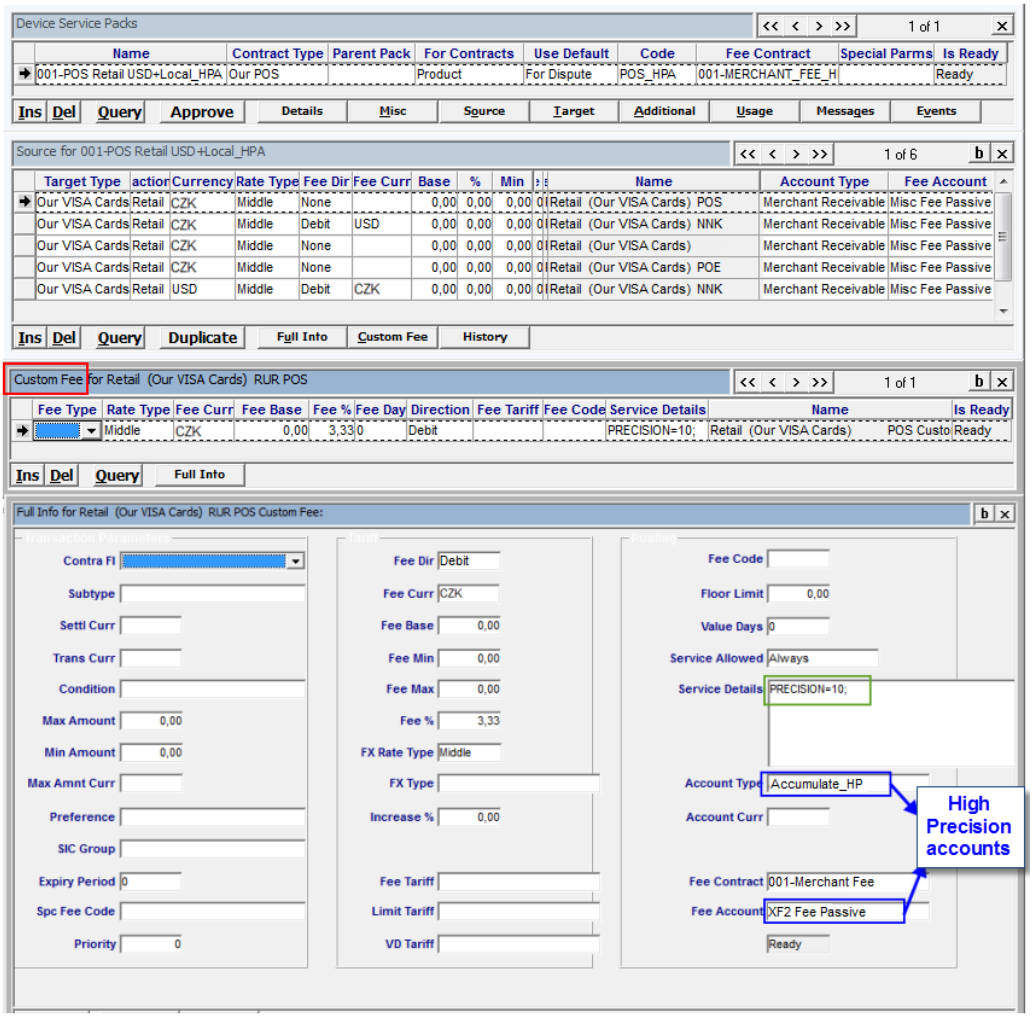


Fig. 17. Settings for a high precision custom fee

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 Fig. 17 of the "Service Settings" section this is the "Accumulate HP" account). This account type is

searched for in the accounts of the main settlement contract for the transaction. I.e. 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 Fig. 17 of the "Service Settings" section). I.e. 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 Fig. 17 of the "Service Settings" section, 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 pair of accounts with the "High Precision" marker is set up (see Fig. 16 in the section "General Scheme of Operation").

"High Precision" account setup (see Fig. 18):

- To specify the "High Precision" marker, select the value "High Precision account" in the GL Type field of the account template.
- 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. I.e when setting up the corresponding account type, "Yes" may not be specified in the *Is Am Av* field (if "Yes" is specified, the Accounting 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.

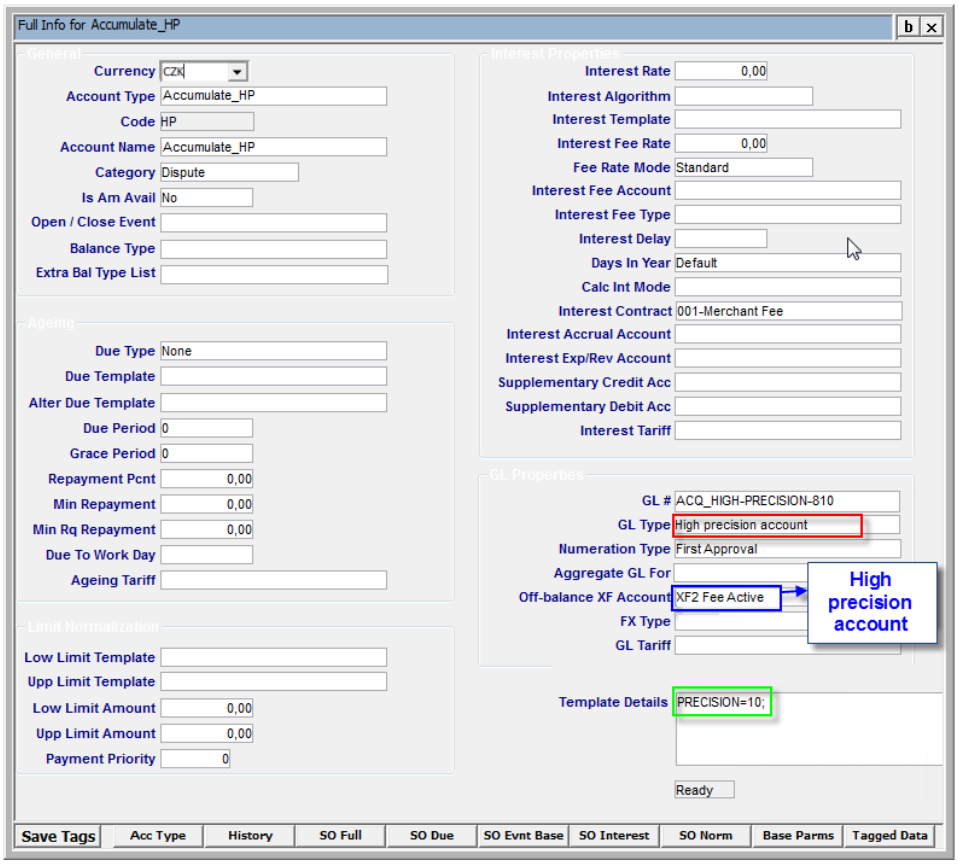


Fig. 18. Template of a merchant contract's "High Precision" cumulative account for recording a fee with high precision

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

In addition to XF accounts explicitly specified in Accounting 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 XF_OWN_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" 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" 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" area (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 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 to determine the account type that will be selected in the XF contract (see the sections "Off-Balance Accounting Subsystem", "Configuring the "GL Types" Form").

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 Fig. 17 in the "Service Settings" section and Fig. 19, 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.

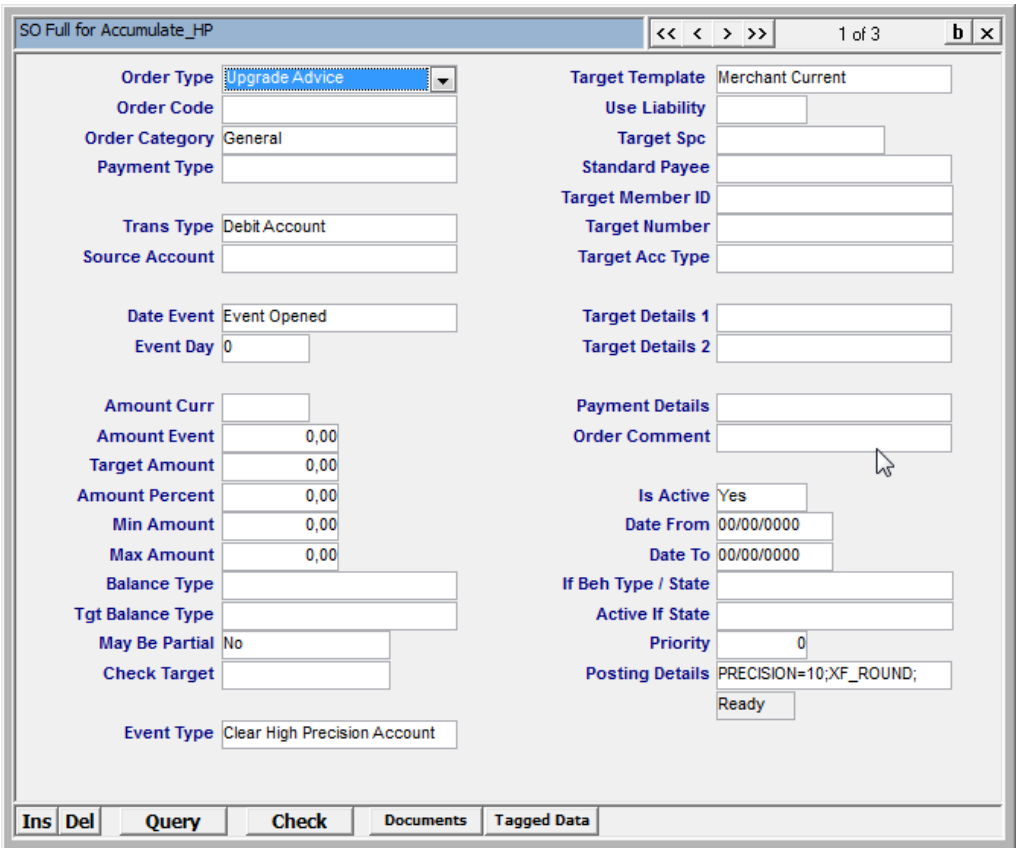


Fig. 19. Setting up a payment order

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).

The template of a standard account is specified as the offset account (*Target Template* field) in an order (XF accounts must be specified in the templates of both accounts).

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 normalisation (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:

1. Debit: HP Fee Active – Credit: Accumulate_HP. 12.13728
2. 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 activate, two macrotransactions are generated:

1. Debit: HP Fee Active – Credit: Accumulate_HP. 12.14 (i.e. a "tail" of 0.00272 remains in the High Precision Fee account)

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

When rounding the remainder in the cumulative "Accumulate_HP" account, the sign may change if the rounded value is greater than the original value, see Fig. 20, Fig. 21.

	Accumulate_HP USD	M	USD	0.00
▸	Accumulate_HP	M	CZK	0.005

Fig. 20. Balance before activation of an order

	Accumulate_HP USD	M	USD	0.00
▸	Accumulate_HP	M	CZK	-0.005

Fig. 21. Change in sign after an order has been activated when rounding up

ⓘ In a case 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).

Other Options for Working with "High Precision" Accounts

The functionality shown below should not be used without consulting the WAY4 vendor and careful testing:

- Limit normalisation between high precision accounts.
- Due normalisation 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 "Setting up a Payment Order (Transferring an Accumulated Amount to a Standard Account)" section). 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 normalisation (with "Account Due" in the *Date Event* field).

- High precision interest accrual. The High Precision marker 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" (i.e. 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.

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. In our example:

- For balance accounting ("Balance" in the *Name* field), set the tag FX_CONV=h;.
- For "High Precision" – FX_CONV=B;.

Example of accounting setup for issuer high precision interchange fee

When acquiring bankcards in a foreign network, payment systems send a high precision interchange fee (with an accuracy of 6 decimal places) in each document. This is information for the issuer about how the payment system will calculate the total acquiring fee. The payment system sends the total amount of revenue/expense for the IPS fee for all documents in a settlement message already rounded to a currency exponent.

The settings described in this section make it possible to do the following:

- Record an high precision interchange fee for transactions with "our" cards on "foreign" devices.
- On a daily basis, designate high precision transactions to the revenue of the corresponding issuer (rounded to two decimal places).
- Makes it possible to balance accounts.

Setup for interchange fee

To record issuer high precision interchange fee, do as follows:

- Add account types for recording the total amount of issuer fees (Full → Configuration Setup → Accounting Setup → Account Types).

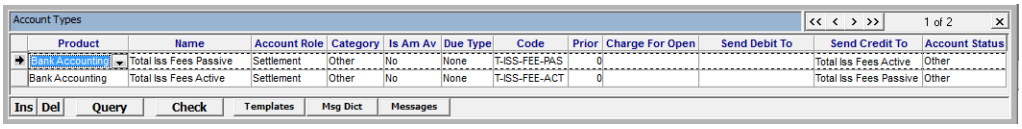


Fig. 22. Form to register and configure account types for recording fees in standard mode

- Add account types for recording high precision fees:

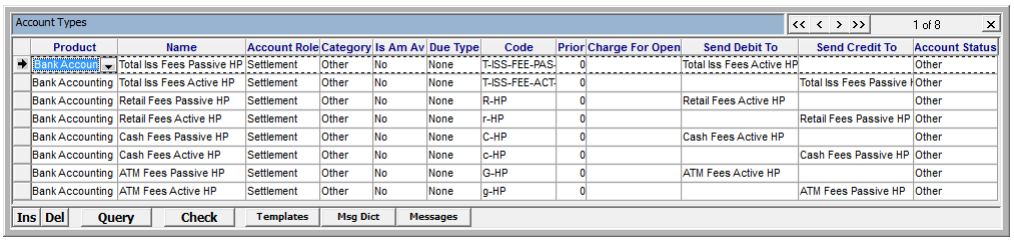


Fig. 23. Configuring account types to record high precision fees

- Add templates for these account types in payment system settlement currencies to Account Schemes for Nostro contracts (Full → Configuration Setup → Accounting Setup → Bank Account Schemes). In the example shown in Fig. 24, account templates are configured for the "001-VISA Nostro" Account Scheme.

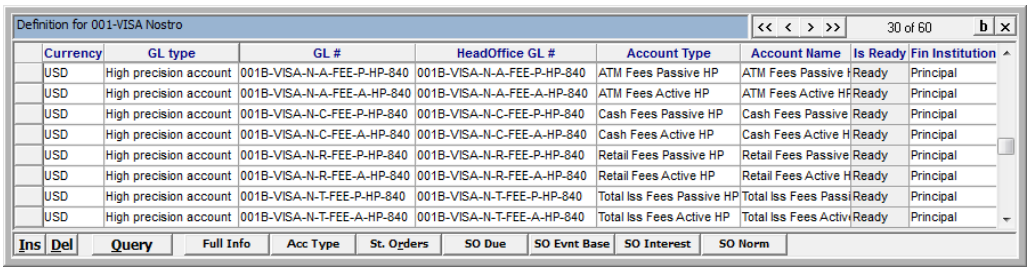


Fig. 24. Configuring account templates in payment system settlement currencies

  - For all the templates that were added, specify "High precision account" in the *GL Type* field.

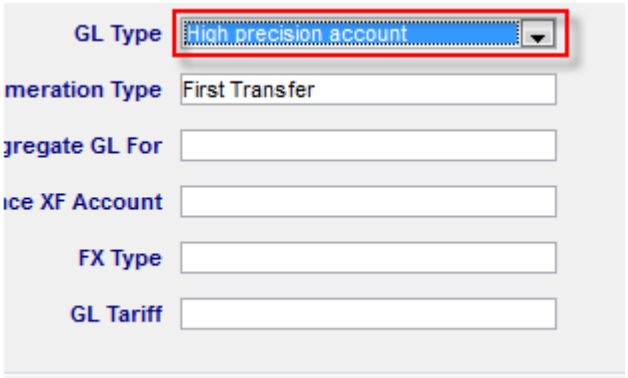


Fig. 25. Setting "High precision account" in an account template

  - For Retail Fees Active HP/Passive HP, Cash Fees Active HP/Passive HP, ATM Fees Active HP/Passive HP accounts, specify the OWN_ALT_XF_ACCOUNT; tag in the *Template Details* field (see Fig. 26). The XF account for the counterparty contract's revenue account (001-Client Fee) will be searched for under the "001-VISA Nostro" account.

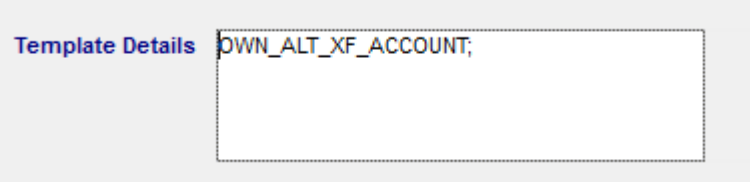


Fig. 26. Setting the OWN_ALT_XF_ACCOUNT in an account template

- In Service Packages for "Nostro" subcontracts with the "Device" category (for example, for VISA these are the 001-VISA Acq and 001-VISA Intracountry Acq subcontracts under the 001-VISA Nostro contract) add a custom fee to Services and make the following settings for the custom fee (see Fig. 27):
  - Specify the tags SRCFD;PRECISION=6; in the custom fee's *Service Details* field.
  - Specify "100" in the *Fee %* field.
  - Select the value "Total Iss Fees Active HP" in the *Account Type* field.
  - Select the value "Retail Fees Active HP" in the *Fee Account* field.

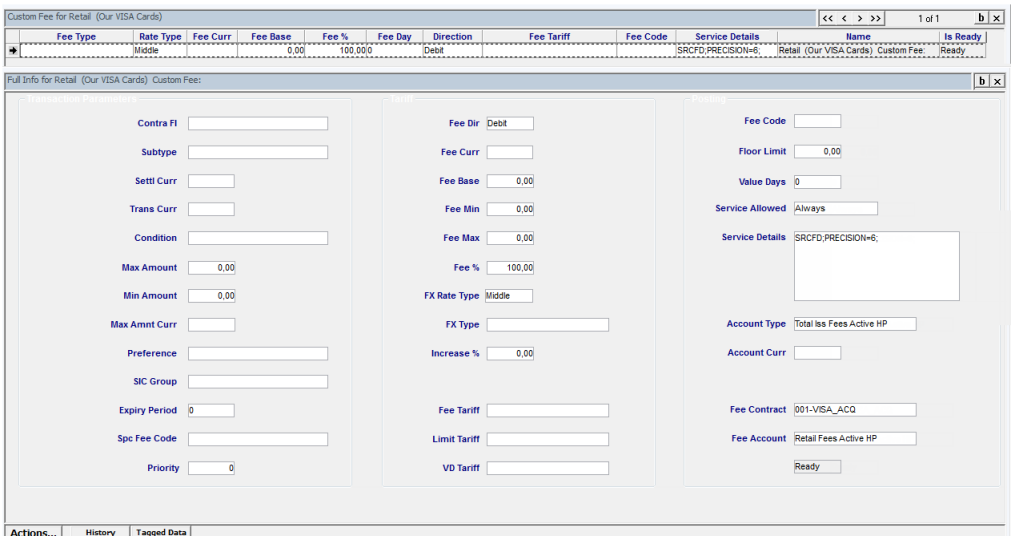


Fig. 27. Configuring a Custom Fee for a "Nostro" subcontract

- When a clearing document with the *Source Fee* field filled with a high precision fee amount is posted for a retail transaction (in the example in Fig. 28, the fee amount contains 6 decimal places), this amount is posted between "High Precision" accounts (see Fig. 29 and Fig. 30).

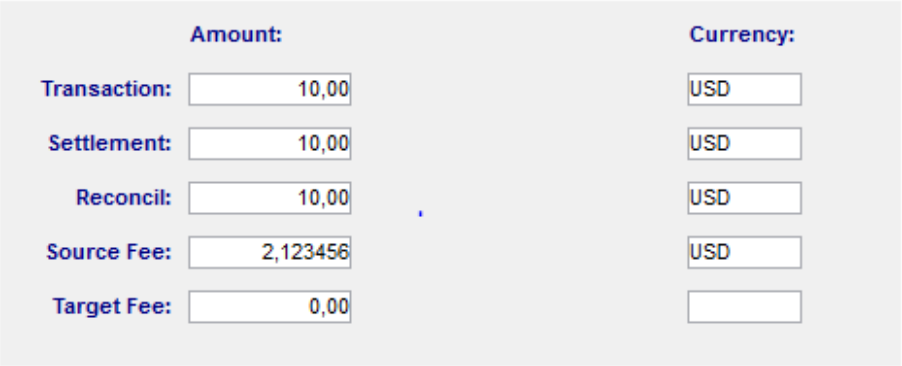


Fig. 28. Parameters of a document according to which a high precision fee is charged

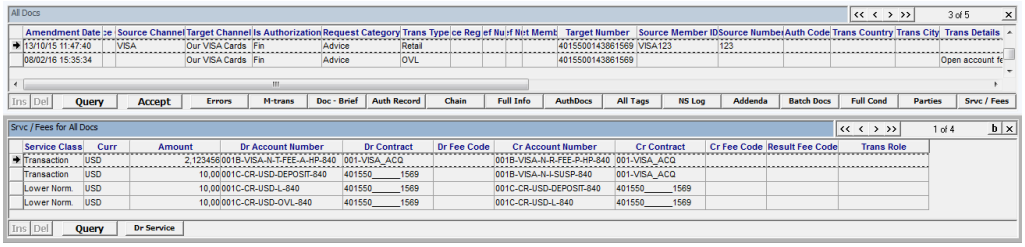


Fig. 29. Entries generated for a document's source contract and target contract accounts

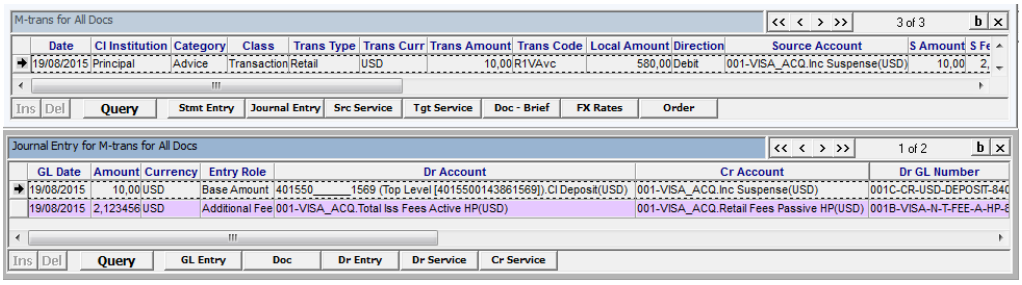


Fig. 30. Journal entries for contract accounts

In Fig. 30 entries are made between the following accounts:

- Debit: Cl Deposit – Credit: Incoming Suspense. For the transaction amount – 10 USD.
- Debit: Total Iss Fees Active HP – Credit: Retail Fees Passive HP. For the high precision fee amount (2.123456).

- Set up a payment order with "Interbranch" in the *Date Event* field to move the fee amount from the "High Precision" cumulative account to the revenue balance account.

  The order is set up in the same account that is specified in the Service for recording this fee (in the "High Precision" cumulative account). In our example, this is the Retail Fees Active HP account (see Fig. 31).

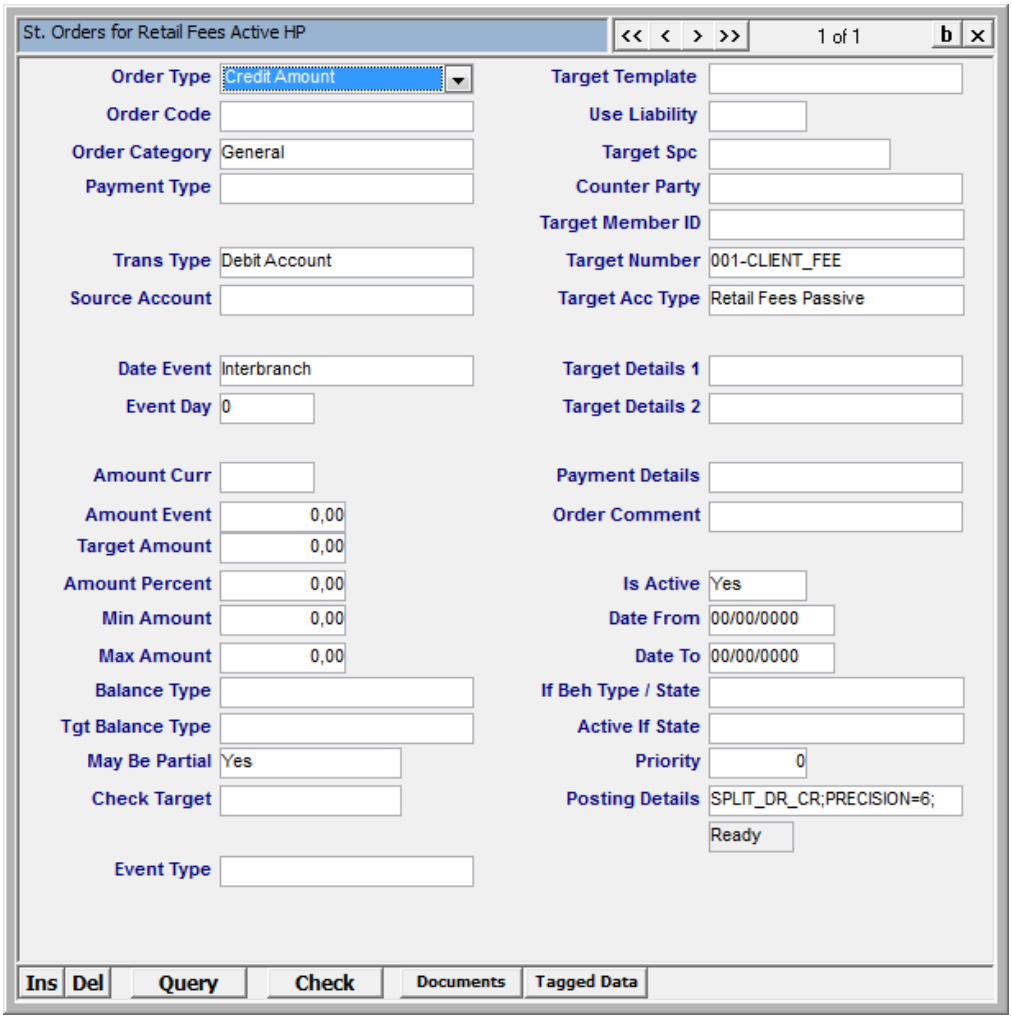


Fig. 31. Payment order for transferring an aggregated fee amount to a revenue account

- Set up a transaction subtype to post the transaction by the payment order with the "Interbranch" attribute (see Fig. 32).

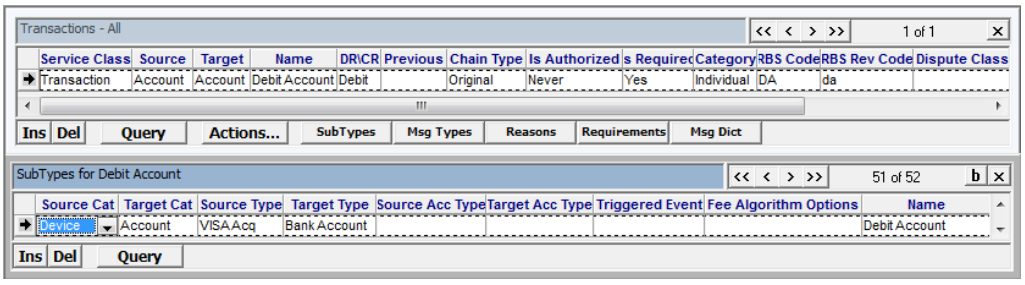


Fig. 32. Transaction subtype to post a transaction by a payment order with the "Interbranch" attribute

- Add a template of an XF account for correspondence with the "High Precision" cumulative account under the Nostro contract when transferring the fee amount from the "High Precision" cumulative account using the order. In our example, an account with the "XF XP" type is added to the 001-Branch Deposit Account Scheme, see Fig. 33.

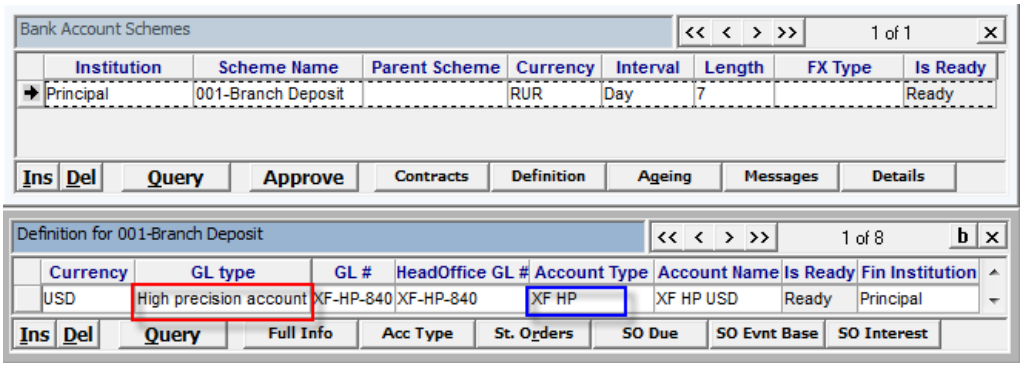


Fig. 33. Transaction subtype

Configure rules for determining the XF account for the "High Precision" account. To do so, set the XF_CONTRACT=FI_DEPOSIT;XF=XF-HP; tags in the *XF Configuration* field for the "High precision account" accounting section in the "GL Types" form (Full → Configuration Setup → Accounting Setup → GL Types) (see Fig. 34). An XF account with this code will be selected from the financial institution's Deposit Contract (i.e. from the 001-Branch Deposit contract).

Specify the PRECISION=6; tag in the *Special Parameters* field (see Fig. 34).

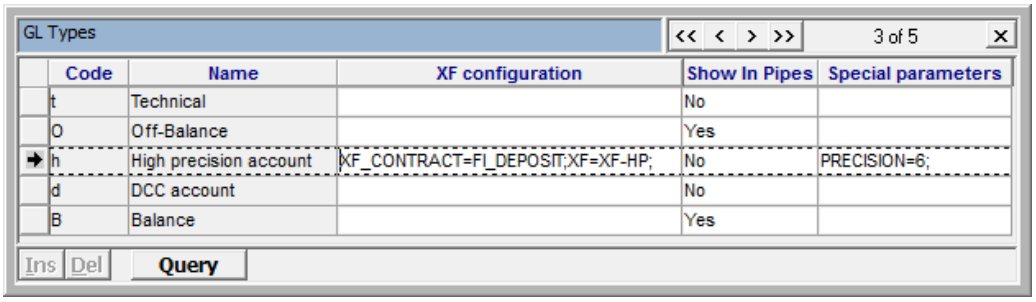


Fig. 34. Setting up the "High precision account" accounting section in the "GL Types" form

- Set up rules for determining XF accounts for correspondence with balance accounts to record fee revenue/expenses (for accounts under the "001-Client Fee" account):
  - For "Retail" revenue/expense accounts, specify the Retail Fees Active account type code – XF-r; in the account template's *Template Details* field, see Fig. 35.

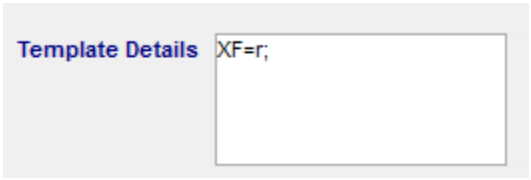


Fig. 35. Setting up definition of the XF account type in the revenue account template

  - Similarly, the Cash Fees Active account type code is specified for "Cash" revenue/expense accounts, and ATM Fees Active account type code for "ATM" revenue/expense accounts.
- When an order is processed, the following entries are generated:
  - Debit: Retail Fees Active HP – Credit: XF-HP. High precision fee amount.

- Debit: Retail Fees Active XF account – Credit: Retail Fees Passive revenue account. Amount rounded to two decimal places.

- Set up transaction subtypes for crediting/debiting a fee (interchange fee) for issuing:
  - Search for transaction types (Full → Configuration Setup → Transaction Types → Transactions – All). For example, for VISA a search may be made according to the criteria shown in Fig. 36.

Fig. 36. Searching for transaction types

  - For the transaction types that were found, filter transaction subtypes for which the *Target Acc Type* field has one of the following values: "Retail Fees Passive", "Cash Fees Passive", or "ATM Fees Passive".

- Configure rules for determining the XF account for the Nostro Suspense account that will be used when transferring reimbursement for the fee from the Nostro Suspense account to the Total Iss Fees Passive HP account. To do so, in the Nostro Suspense account template, specify the Total Iss Fees Passive account code and the OWN_XF_ACCOUNT; tag in the XF tag (see Fig. 39). Movement of funds from the Nostro Suspense account to the Total Iss Fees Passive HP account when posting a clearing document with settlement information is determined by the accounts defined in the transaction subtype.

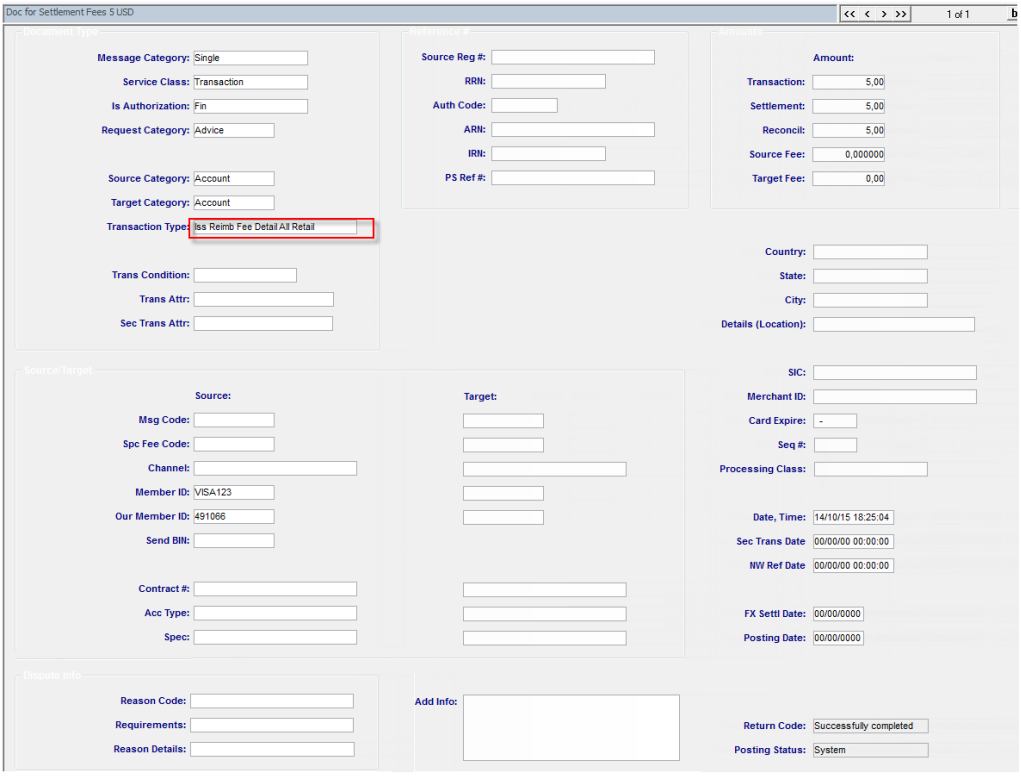


Fig. 37. Example of a fee reimbursement document

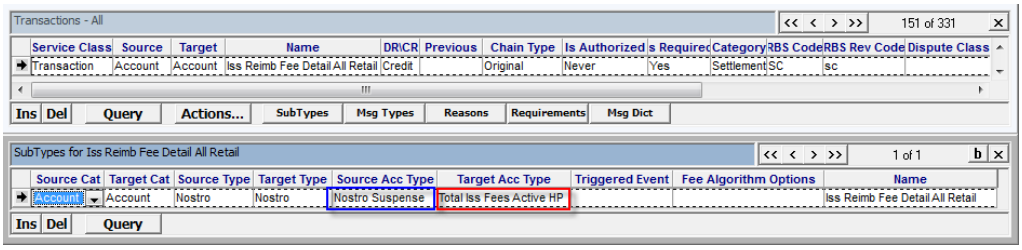


Fig. 38. Transaction subtype for reimbursement

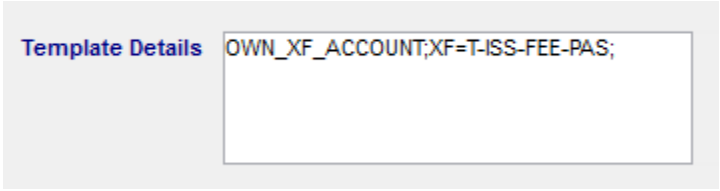


Fig. 39. Rules for determining an XF account for a Nostro Suspense account

When the document from the payment system to reimburse the fee is posted, the following entries are generated:

- Debit: Nostro Suspense – Credit: Total Iss Fees Passive
- Debit: XF-HP – Credit: Total Iss Fees Passive HP

- Set up an order for daily consolidation of all received/debited fees in the Total Iss Fees Passive/Active account. Orders are set up in Retail Fees Active/Passive, and/or Cash Fees Active/Passive and/or ATM Fees Active/Passive accounts:
  - An example of an order in an active account is shown in Fig. 40.

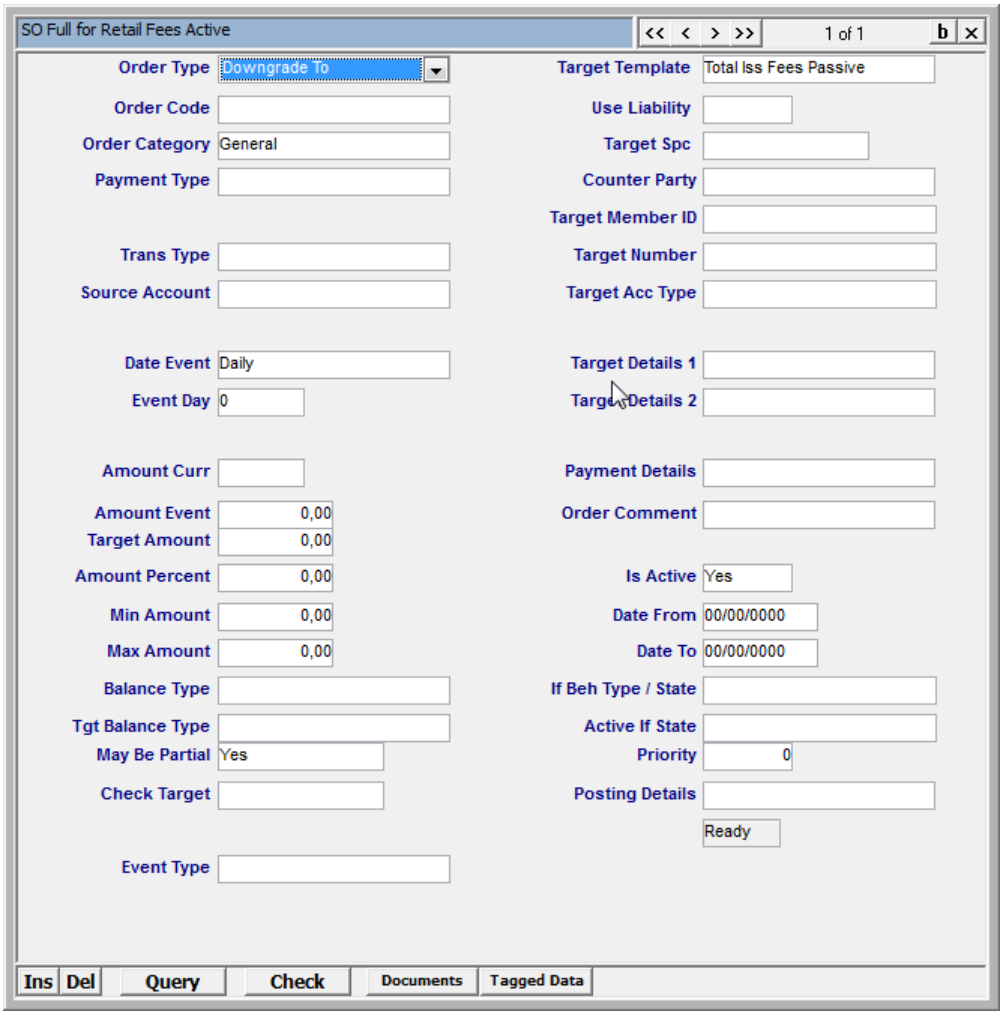


Fig. 40. Rules for determining an XF account for a Nostro Suspense account

- An example of an order in a passive account is shown in Fig. 41.

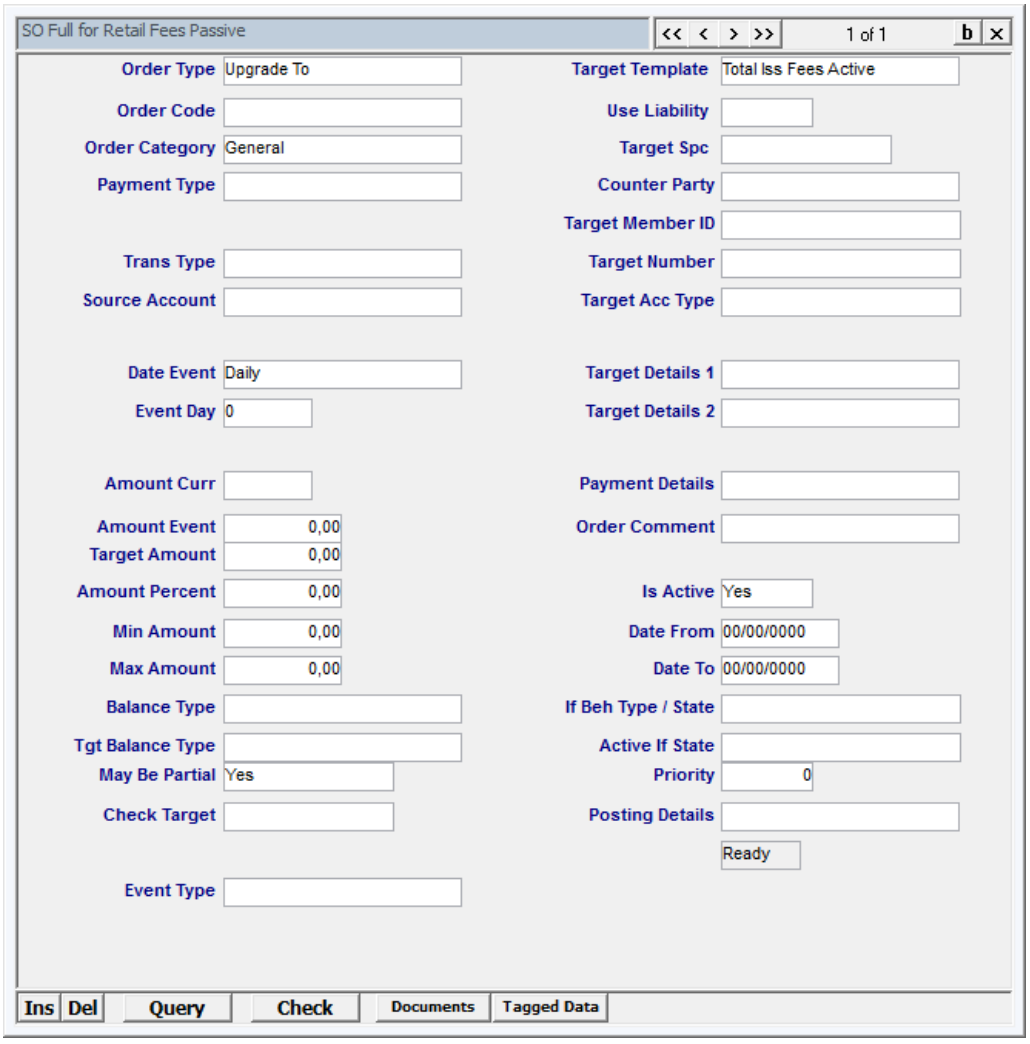


Fig. 41. Rules for determining an XF account for a Nostro Suspense account

Entries resulting from setup

Table 3. Accounts used

Account type name	Contract	Description
Inc Suspense	NOSTRO	Account for showing incoming clearing documents from the IPS.
Nostro Suspense	NOSTRO	Account for consolidating clearing documents with settlement information from the IPS.
Total Iss Fees Passive/Active	NOSTRO	Active/passive account pair for recording the total fee amount received/debited for card transactions.
Fees Passive/Active	NOSTRO	Active/passive account pair for recording the fee amount received/debited for card retail transactions.
ATM Fees Passive/Active	NOSTRO	Active/passive account pair for recording the fee amount received/debited for ATM card transactions.
Cash Fees Passive/Active	NOSTRO	Active/passive account pair for recording the fee amount received/debited for cash dispensing transactions.

Total Iss Fees Passive/Active HP	NOSTRO	Active/passive account pair for recording the total high precision fee amount received/debited for card transactions.
Retail Fees Passive/Active HP	NOSTRO	Active/passive account pair for recording the high precision fee amount received/debited for card retail transactions.
ATM Fees Passive/Active HP	NOSTRO	Active/passive account pair for recording the high precision fee amount received/debited for ATM card transactions.
Cash Fees Passive/Active HP	NOSTRO	Active/passive account pair for recording the high precision fee amount received/debited for cash dispensing transactions.
XF-HP	DEPOSIT	Bucket account for posting high precision transactions (common account for correspondence with "High precision" accounts)
Retail Fees Passive/Active	CLIENT_FEE	Bank revenue/expenses for retail transactions.
ATM Fees Passive/Active	CLIENT_FEE	Bank revenue/expenses for ATM transactions.
Cash Fees Passive/Active	CLIENT_FEE	Bank revenue/expenses for cash dispensing transactions.

Table 4. Account entries

Name	**Date**	**Accounting**		
		Debit	**Credit**	**Amount**
Posting a clearing document (Retail)	On the document posting date	Customer Account (Cl Deposit)	NOSTRO. Inc Suspense	Transaction amount
		NOSTRO. Total Iss Fees Active HP	NOSTRO. Retail Fees Passive HP	High precision interchange fee amount
Receiving a clearing document from the IPS for reimbursement of the interchange fee (Settlement Total)	On the document posting date	NOSTRO. Nostro Suspense	NOSTRO. Total Iss Fees Passive	Amount of reimbursement from the payment system
		DEPOSIT.XF-HP	NOSTRO. Total Iss Fees Passive HP	
Assigning revenue/expenses for transactions with the IPS (Interbranch order)	On the document posting date	NOSTRO.Retail Fees Active HP	DEPOSIT.XF-HP	Amount of all Retail high precision interchange fees
		NOSTRO. Retail Fees Active	CLIENT_FEE.Retail Fees Passive	Rounded amount.
Consolidation of all credited/debited fees in the Total Fees account at	At the end of the day	NOSTRO. Total Iss Fees Active	NOSTRO. Retail Fees Passive	Total amount of Retail fees

Name	Date	Accounting		
		Debit	Credit	Amount
the end of the day				

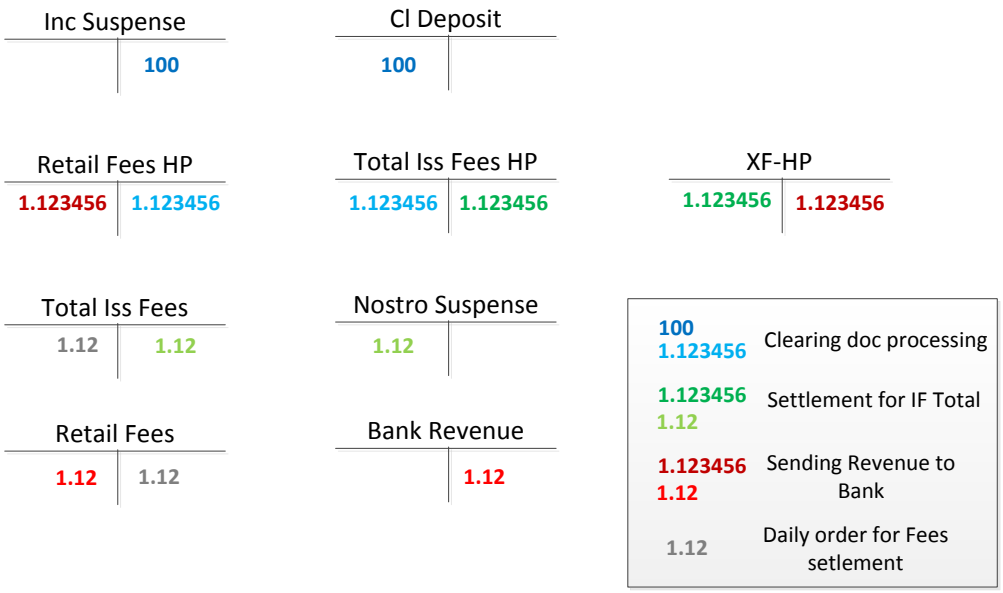


Fig. 42. Illustration of balancing accounts