Operation Manual

Products and Contract Subtypes

03.52.30

21.07.2021

Contents

This document is intended for bank or
processing center employees who are responsible for configuring Way4 and describes Product creation and configuration.
When working with this document, it is recommended to use the following resources from the Way4 documentation series:

- Advanced Applications R2
- Way4 Account Schemes
- Way4 Service Packages
- Documents
- Way4 Global Parameters
- Issuing Module
- Acquiring Module
- Way4 Advanced Tariff Management
- Preferred Counterparties
- Way4 Client and Contract Classifiers

The following notation can be used in the document:

- Field labels in screen forms are shown in *italics.*
- Key combinations are shown in angular brackets, for example, <Ctrl>+<F3>.
- Names of screen form buttons and tabs are shown in square brackets, for example, [Approve].
- Sequences for selecting user menu items or context menu items are shown using arrows as follows: "Issuing → Contracts Input & Update".
- Sequences for selecting system menu items are shown using arrows as follows: Database => Change password.
- Variables that differ for each local instance, such as directory and file names, as well as file paths are shown in angular brackets, as in <OWS_HOME>.

Warnings and information are marked as follows:

Warnings about potentially hazardous situations or actions.

Messages with information about important features, additional options, or the best use of certain system functions.

1 Terms and Definitions

Way4 is used to resolve the following tasks:

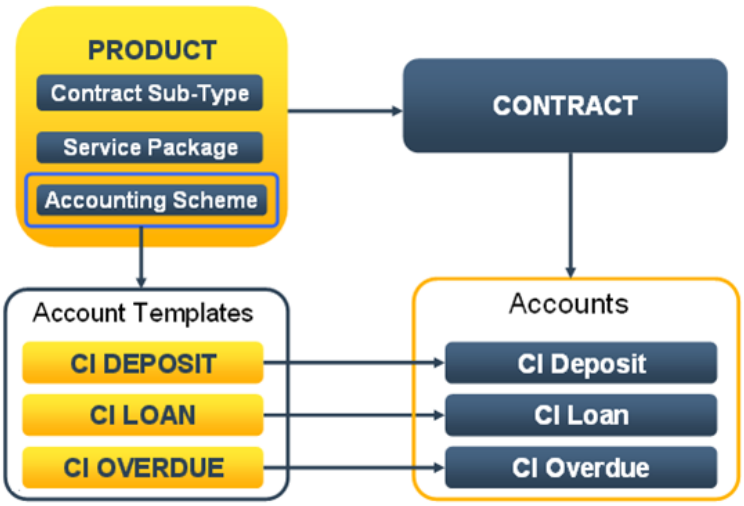
- Issue and acquire payment cards of individuals and corporate clients.
- Acquire merchants.
- Account and process financial transactions, including card transactions.
- Service current, deposit and loan accounts of individuals.

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

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

Contract properties are determined by three main parameters:

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



Relations between system objects

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

2 Products

A Product is used as a template to register new contracts or to change properties of contracts already registered in Way4 using Products.

With the Way4 Products users can:

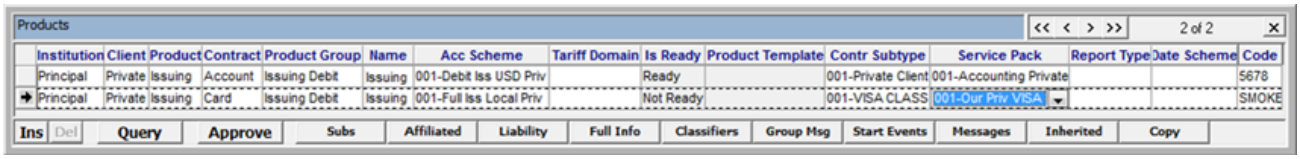
- Specify standard groups of contract properties.
- Create a hierarchical contract tree according to product characteristics.

Information about Products is used in Issuing and Acquiring Modules when contract data is entered and in the Advanced Applications module (see the document "Advanced Applications R2") when applications to issue cards are entered.

2.1 Entering Product Data

2.1.1 Main Product Parameters

To create a new product, select "Full → Configuration Setup → Products → Product Definition → Products" from the user menu. As a result, the "Products" form will be displayed (see figure).



Form for entering Product data

Products are classified as follows:

- *Institution* – financial institution registered in Way4 for which the Product is created.
- *Client* – determines the client type:
  - "Private" – an individual.
  - "Commercial" – legal entity.
  - "Accountant" – bank division.
- *Product* – drop-down list of Product categories ("Issuing", "Acquiring", "Bank Accounting", or "Accounting" (the last value is only used for backward compatibility)).
- *Contract* – Contract category for which the Product may be used (Card, Account, Device).
- *Product Group* – name of a Product group registered in the system (see the section "Creating Product Groups"); used in the Advanced Applications module.
- *Name* – Product name that will be used to create contracts.
- The *Tariff Domain* field for each Product may contain one of tariff domains registered in the system. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module. For more information, see the document "Way4 Advanced Tariff Management".

> The tariff management module is not included in the basic configuration of Way4 and is provided according to an additional agreement with OpenWay.

- *Product Template* – name of the template Product (see the section "Copying Products to another Financial Institution with Inheritance of Properties").
- The *Is Ready* field shows whether changes made to the Product have been approved.
- *Code* – Product code used when files from external sources are imported into the system to link imported data with a specific Product.

Product components:

- *Accounting Scheme* – Account Scheme that determines a set of accounts and the way they interact (see the document "Way4 Account Schemes").
- *Contract Subtype* – contract subtype (see the section "Contract Types and Subtypes").
- *Service Package* – Service Package that determines a set of authorized operations, Product fees, etc. (see the document "Way4 Service Packages").
- *Report Type* – report type; to configure a report type, select "Full → Configuration Setup → Products → Reporting → Report Types" from the user menu. The report type is used to mark and subsequently group contracts when generating reports (a report will be generated for all contracts with this report type).
  During Product setup, the *Report Type* field is not mandatory since a report type can be specified on the contract level.
  The forms "Report Templates" (Full → Configuration Setup → Products → Reporting → Report Templates) and "Report Item" (Full → Configuration Setup → Products → Reporting → Report Types → [Report Item]" are reserved for forward compatibility and are not used in standard reports.
- *Date Scheme* – scheme for calculating contract functional dates (see the document "Contract Functional Dates").

The [Approve] button contains the following context menu commands:

- "Approve" – approve a Product. See the section "Approving Products and Applying Product Changes to Contracts".
- "Apply" – update the properties of contracts and their accounts registered for a Product according to changes in the Product (performed when the value of the global parameter APPROVE_IMMEDIATE is "No").
- "Fill Down Date Scheme for Children" – inherit date scheme parameters for subordinate Products. For "Main/Sub" hierarchies, no inheritance takes place as the date scheme configured in the main contract is used by default.

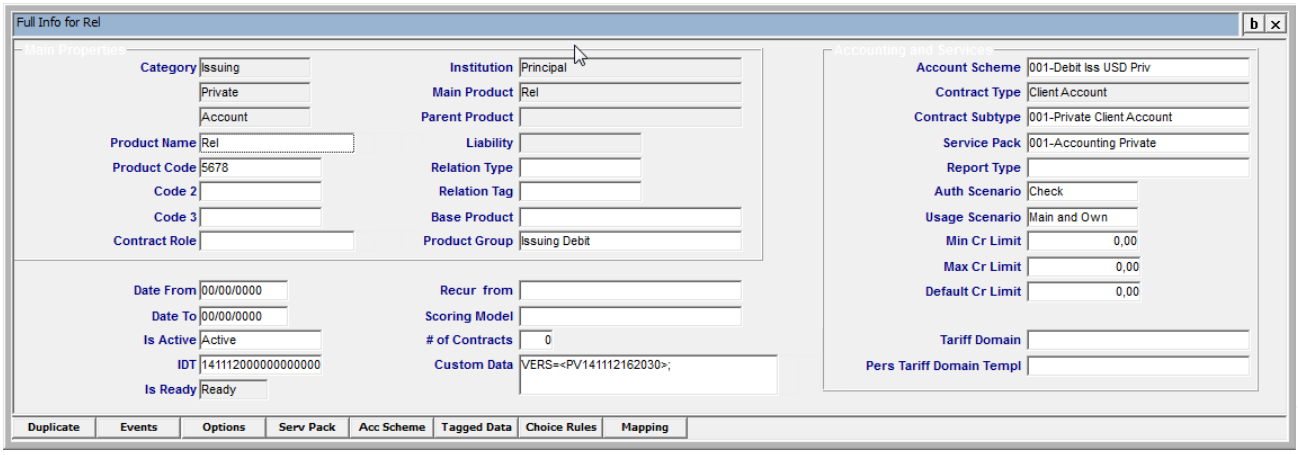
> If a date scheme is already set for a subordinate Product, when the "Fill Down Date Scheme for Children" menu item is executed, the subordinate Product's date scheme is updated according to the main Product's settings. A warning message is recorded in the process log (Full → Process Log).

Buttons [Subs], [Affiliated] and [Liability] in the form are used to set up a Product hierarchy (see the section "Configuring Product Hierarchies").
The [Copy] button is used to copy a Product to another financial institution (see the section "Copying Products to another FI").
The [Full Info] button is used to access the form with additional Product information (see the section "Additional Product Parameters").
The [Classifiers] button is used to set the values of contract classifiers that will be assigned when a contract for this Product is created (see the section "Default Contract Classifiers for a Product").
The [Group Msg] button is used to set up client message templates (see the section "Configuring Message Templates (Group Msg)").
The [Start Events] button is used to access the form for specifying Events that will be opened when a contract using this Product is opened or closed (see section "Event Setup").

> An unused Product or Product that was created incorrectly is deleted using a special menu item in a separate form. Rules for deleting a Product are described in the section "Deleting a Product" of the document "Special Contract Utilities".

2.1.2 Additional Product Parameters

To access additional Product parameters, click the [Full Info] button in the "Products" form (see the figure in section "Main Product Parameters"). This will open the "Full Info for <name of Product>" form (see figure) used to specify additional Product parameters.



Form for entering additional Product parameters

The form contains the following fields absent from the parent form:

- *Code 2*, *Code 3* – additional Product codes.
- *Contract Role* – contract role. The value is selected from a list configured in the "Contract Roles" form (Full → Configuration Setup → Products → Product Definition → Contract Roles), see the section "Configuring the "Contract Roles" List".
- *Date From*, *Date To* – dates for determining the time interval during which the Product is active. An active Product can be used to create a new contract or to change the Product of an existing

contract. If a Product is inactive (the current banking date is not within the specified time interval), this has no impact on operation of existing contracts that use this Product. When users attempt to use an inactive Product to register a new contract (or to change the Product of an existing contract), an error message is displayed.

- *Is Active* – when the field is set to the "Active" value, the Product is active; when the field is set to the "Closed" the Product is inactive. See a definition of active and inactive Products in the description of fields Date From, Date To.

> Users can specify a Product status (active/inactive) in two ways: using fields *Date From*, *Date To* or using the *Is Active* field.

- *IDT* – internal Product code.
- *Main Product* – main Product of the current Product. A main Product is a Product found on the highest level of a Product hierarchy using "Main/Sub", "Affiliated Main/Sub" and "Liability Main/Sub" Product relations.
- *Parent Product* – parent Product. A parent Product is a Product found on the next level up of a Product hierarchy using "Main/Sub", "Affiliated Main/Sub" and "Liability Main/Sub" Product relations.
- *Liability* – type of relation between the current Product and its parent Product in a "Liability" hierarchy (if the Product is not linked to a parent Product with relations of the "Affiliated Main/Sub" or "Liability Main/Sub" type, the field is left blank). For more information, see the section "Configuring Product Hierarchies".
- *Relation Type* – type of relation between card contracts or device contracts. If this field is filled in, this Product can be selected when registering related contracts. For more details, see the section "Related cards" of the document "Issuing Module", and the section "ATM Retail Contracts" of the document "Acquiring Module". The field value is selected from a drop-down list of relation types registered in the system (Full → Configuration Setup → Accounting Setup → Contract Relations).
- *Relation Tag*:
  - When the "Active Related" value is selected, the relation type specified in the *Relation Type* field is activated.
  - When the "Inactive Related" value is selected, the relation type specified in the *Relation Type* field becomes inactive (the relation is broken).
  - When the "Applet" value is selected, this Product can be selected during registration of additional card application (Applet) contracts.
- *Base Product* – the field is filled in for Products used to register related contracts or additional card applications (Applets). The *Base Product* field specifies the main contract's Product or the main applet's Product. This field is not filled in for the main contract's (applet's) Product.
  The relation between Products through the *Base Product* field makes it possible to limit the creation of additional applets and related contracts. For example, a related contract (created on the basis of a Product with the *Base Product* field filled in) can only be "attached" to the contract whose Product is specified in the *Base Product* field. This limitation is used when automatically generating contracts.

- *Recur from* – parent Product whose properties must be inherited (Product characteristics may be inherited by sub-products from main Products). For more information, see the section "Inheritance of Product Properties".
- *# of Contracts* – filled in for a subordinate Product in a "Main/Sub" hierarchy. If the value in this field is "1", a subcontract will automatically be created when the contract for the main Product in the "Main/Sub" hierarchy is approved.
- *Custom Data* – field for entering additional parameters as tags. For more details, see the section "Tags used when configuring Products and contract subtypes" of the document "Setup Tags". By default, this field contains a tag showing a Product version number automatically generated every time a Product is approved.
- *Scoring Model* – the field is used for backward compatibility.
- *Contract Type* – contract type corresponding to the selected contract subtype.
- *Auth Scenario* – algorithm used to calculate the amount available during authorization (for more details, see the section "Authorization Scenarios").
- *Usage Scenario* – the value of this field determines how usage limiters are applied during authorization request processing (for details, see the document "Usage Limiters"):
  - "Main and Own" – the system checks the limiters set up for both the contract using this Product and its parent contract.
  - "Own only" – the system only checks the limiters set up for the contract using this Product.
- *Max Cr Limit, Min Cr Limit* – maximum and minimum contract credit limits, respectively.
- *Default Cr Limit* – default credit limit automatically set for contracts using this Product.
- *Pers Tariff Domain* – in this field, an individual tariff domain can be specified for each Product. A value can be selected in this field if the distribution package includes the Advanced Tariff Management module. For more information, see the document "Way4 Advanced Tariff Management".

> The tariff management module is not included in the basic configuration of Way4 and is provided according to an additional agreement with OpenWay.

The [Duplicate] button is used to copy a Product (see the section "Copying Products").

The [Serv Pack] button is used to access this Product's Service Package.

The [Acc Scheme] button is used to access this Product's Account Scheme.

The [Tagged Data] button is used to optimize the use of the *Custom Data* field. Clicking this button opens a grid form for entering and editing tags specified in the *Custom Data* field (see the section "Entering and Editing Tags").

The [Choice Rules] button makes it possible to set rules for selecting Products that will be available to a client. See the section "Configuring Available Products".

The [Mapping] button is used to mark Products with "Configuration Groups" classifiers. For more information, see the section "Default Contract Classifiers for a Product".

The [Options] button remains for backward compatibility. In earlier versions (up to 03.38.19.11) the value was used when configuring Product options. It is recommended to set up this functionality with classifiers (see the section "Default Contract Classifiers for a Product").

2.2 Configuring Product Hierarchies

Besides storing sets of contract standard properties, Products also define the contract hierarchy. This enables the development of standard hierarchy structures for mass products using basic types of contract relationship.

Clicking the [Subs], [Affiliated] or [Liability] control buttons at the bottom of the "Products" form (see the figure in section "Main Product Parameters") creates sub-products connected to the current main product according to a "Main/Sub", "Affiliated Main/Sub" or "Full Liability Main/Sub" relationship, respectively.

After a button is clicked, the form specifying sub-product properties appears. The form fields are similar to the "Products" form fields and are filled according to bank policies.

> Relation types "Only Check Balance Main/Sub" and "Reporting Main/Sub" used in "Liability" contract hierarchies do not need to be supported on the Product hierarchy level.

When a child Product with the "Main/Sub" relation type is created, it must use the same Account Scheme as the parent Product. The Account Scheme is automatically copied to the child Product's properties. When users attempt to change the Account Scheme, a warning message is generated.

For more details on contract relation types, see section "Contract Hierarchies" of the document "Issuing Module".

2.3 Copying Products

Way4 allows Products to be copied. This function allows users not to make a great number of configurations when it is necessary to create a new Product based on an existing one.

2.3.1 Copying Products within a Financial Institution

To copy Products within the same FI, do as follows:

- Click the [Duplicate] button in the additional Product data form (see the figure in section "Additional Product Parameters"). The form "Duplicate for <Product name>" will open.
- In the "Duplicate for <Product name>" form, click the [Duplicate] button and execute a context menu command:
    - "Copy" – standard mode for copying Products. In this mode, the selected Product and all its sub-products are copied (the entire Product hierarchy is copied). If tariff domains are set in

the Products being copied, duplicate Products will refer to the same tariff domains as the original Products. In the copied hierarchy, the copying date and time are added to the name of the top-level Product. The names of copied sub-products do not change.

- "Copy Product and Tariff Domain" – when copying a Product hierarchy, the corresponding tariff domain hierarchy is copied, beginning with the tariff domain level set in the properties of the top-level Product. In this mode, Product copies refer to the corresponding tariff domain copies. The postfix "(New)" is added to the name of the copied top-level tariff domain as well as the copying date and time. Names of copied subordinate domains do not change.

  > The tariff management module is not included in the basic configuration of Way4 and is provided according to an additional agreement with OpenWay.

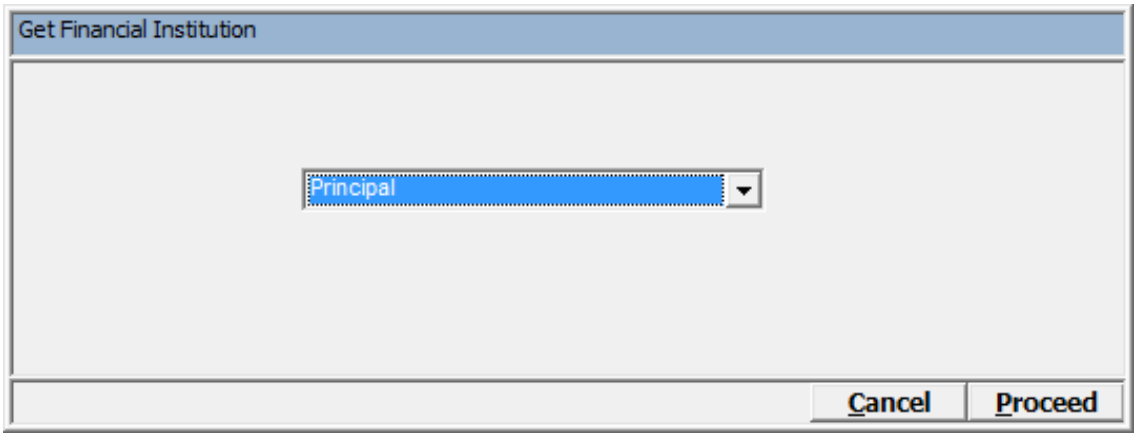
- After copying, a new Product is added in the list of Products in the "Products" form (see the figure in section "Main Product Parameters").

  It is necessary to change the parameters of the copied Product and approve the Product.

2.3.2 Copying Products to another FI

To copy a Product to another FI, proceed as follows:

- In the "Products" form (see the figure in the section "Main Product Parameters"), select the required Product and click the [Copy] button.
- In the "Copy for < Product name>" dialogue box that opens, click the [Do...] button and execute the context menu item "Copy Product to Institution".
- In the "Get Financial Institution" form that opens (see figure), select the FI to which the Product will be copied and click the [Proceed] button.



Form for selecting an FI

If when copying a Product it turns out that the target financial institution already has a Product with this code (the *Product Code* and *Code 2Code 3* fields are checked), components of the target financial institution's Product will be replaced with those of the source financial institution.

When copying a Product to another financial institution, the contract subtype's *RBS Code* field is updated according to the new prefix (according to the prefix of the new financial institution). Usually, the *RBS Code* field contains a value in the format <financial institution prefix>-<subtype code>. If the *RBS Code* field does not contain a value in digits, the field will not be updated. In this case, it is assumed that the *RBS Code* field value is set in a different format and does not contain a financial institution prefix.

Starting from version 03.45.10, if the main Product to which new subordinate Products are being copied does not change, its status remains "Ready ". I.e. approval is required only for new (and changed) subordinate Products – the version of the main Product and unaffected subordinate Products does not change.

2.3.3 Copying Products to another Financial Institution with Inheritance of Properties

To make it easier to synchronize parameter changes in Products belonging to different financial institutions, it is possible to make changes to one template Product with the subsequent automatic change in parameters of the other Products linked to it. This is true for the template Product's entire hierarchy, i.e. changes will be inherited both from the main Product in the hierarchy and for all subordinate Products.

To copy a template Product to another financial institution, do as follows:

- In the "Products" form (see the figure in the section "Main Product Parameters"), select the required Product (main Product in the hierarchy) that will be used as the template, and click the [Copy] button.
- In the "Copy for <Product name>" dialogue box that opens, click the [Do...] button and execute the context menu command "Link Product to Institution".
- In the "Get Financial Institution" form that opens (similar to the form in the figure of the section "Copying Products to another FI") select the financial institution to which the Product will be copied and click the [Proceed] button.

When copying to the selected financial institution, a Product (Product hierarchy) is created that inherits the properties of the original Product (Account Scheme and Service Package parameters, classifiers, etc.). For the main Product, the *Product Template* field is automatically filled in with the name of the template Product. The *Product Template* field is not filled in for subordinate copied Products.

After changes affecting the template Product or subordinate Products in the hierarchy of the template Product are approved in the database, information is automatically shown in linked Products of other financial institutions (i.e. when there are links to a template Product in the *Product Template* field of the main Product).

If the properties of a subordinate Product in a hierarchy should not be inherited, set the NO_COPY tag in this linked Product. When this tag is specified, the link is broken between the subordinate Product and the corresponding template Product, i.e. after setting this tag and approving changes, the parameters of the template subordinate Product will not be copied to this Product.

2.4 Inheritance of Product Properties

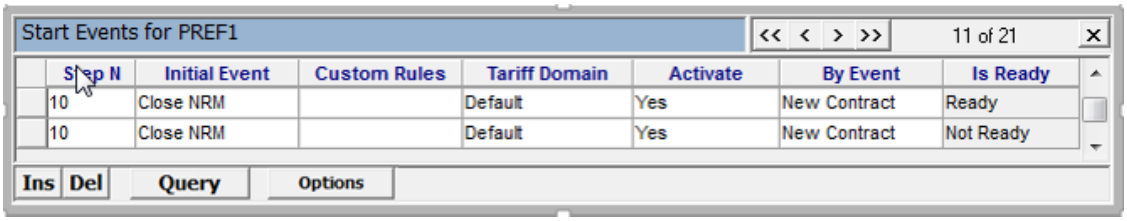
Product characteristics may be inherited by sub-products from main products. For this, the *Recur From* field in the child Product's additional data form "Subs for <name of Product>" is used. It contains a drop-down list of upper-level Products.

> When the *Recur From* field is used, all Product properties are inherited (except Product status and *Date From*/*Date To* dates). It is not recommended to use this field without consultation with OpenWay.

If wrong data is entered, this will interrupt activation and display an error message.

2.5 Event Setup

Events for contracts using this Product are set up in the "Start Events for <name of Product>" form (see figure). To access the form, click the [Start Events] button in the additional Product data form.



Event setup

To add an Event, click the [Ins] button in the above form and fill in the fields of the new record:

- *Step N* – this field is used to specify the order of Event processing. Numeric values (0, 1, etc.) are specified in this field. An Event with value Step N="1" will be processed before the Event with value Step N="0".

  > The order of Event processing is specified for a single Event opening condition (field *By Event*), e.g. the order of processing Events when a contract is opened (By Event="New Contract").

- *Initial Event* – drop-down list of Event types registered in the "Event Types" form (for more details, see the document "Events").
- *Custom Rules* – field for entering additional parameters.
  For example, IF_CS group tags for setting the dependence of an Event opening on client and contract classifier values can be specified in this field. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "Way4 Client and Contract Classifiers".

- *Tariff Domain* – used to select a tariff domain registered in Way4.
- *Activate* – this field allows activation ("Yes") or deactivation ("No") of a tariff domain by an Event. The *Tariff Domain* and *Activate* fields allow activation/deactivation of tariff domains for contracts created on the basis of a given Product at a certain point in the lifecycle of these contracts.
- *By Event* – condition for opening the Event:
  - "New Contract" – the Event is opened automatically when a contract using the Product is opened. A contract is considered opened when it is successfully approved for the first time, and the "Ready" value is specified in the *Approval* field of the contract.
  - "Close Contract" – the Event is opened automatically when a contract using the Product is closed. A contract is considered closed when a closing date is set for the contract, and the current banking date is later than the specified closing date (for more details, see the document "Issuing Module"). In this case, the "Closed" value is specified in the *Approval* field of the contract.
  - "Lifecycle" – this condition is used to change a contract's tariff domain at a specific step of its lifecycle. The Event is set up for the contract and opened in one of the standard ways (see the document "Events"). Event opening only changes a tariff domain when a domain code is specified on the Product level.

    > The tariff management module is not included in the basic configuration of Way4 and is provided according to an additional agreement with OpenWay.

  - "Product Option (Obsolete)" – this value remains for backward compatibility. In earlier versions (up to 03.38.19.11) the value was used when configuring Product options. It is recommended to set up this functionality with classifiers (see the section "Default Contract Classifiers for a Product").
    Note that when the old approach for Product option setup is used, separate Event types must be configured:
    - If the "Product Option (Obsolete)" condition for opening an Event is set, when the Product is approved, the PRODUCT_OPTION; tag will be specified in the *Special Parms* field of the corresponding Event type.
    - An Event type with the PRODUCT_OPTION; tag cannot be used in the "Start Events" form for configuring Events with other conditions for opening (with the "New Contract"/"Close Contract" value in the *By Event* field; this Event will not be opened when the contract is opened/closed). I.e. the same Event cannot be set for Product1 with the "New Contract" value and for Product2 with the "Product Option" value in the *By Event* field. In this case, two Event types must be configured. This limitation extends to Event types in one financial institution.

After filling in the form fields, click the [Approve] button in the "Products" form to approve the changes.

The [Options] button remains for backward compatibility. In earlier versions (up to 03.38.19.11) the value was used when configuring Product options. It is recommended to set up this functionality with classifiers (see the section "Default Contract Classifiers for a Product").

2.6 Entering and Editing Tags

The "Full Info for name of Product" form (see the figure in the section "Additional Product Parameters") contains special field *Custom Data* for entering and editing tags.

The "Tagged Data" form (see figure) is used to optimize the process of entering and editing tags in the above field. The form is opened by clicking the [Tagged Data] button in the "Full Info for name of Product" form.

"Tagged Data" form

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

- A tag name in the *Tag* field can be selected from the system list of tags. If a tag is absent from the list, its name can be entered from the keyboard.
- In the *Value Data* field, enter a tag value.
- The *Value Tag* field:
  - When entering tag parameters in the "Tagged Data" form, select the "Tag Present" value in the *Value Tag* field so that tag parameters are saved correctly. After data is saved, tag parameters will be displayed in the *Custom Data* field in the "Full Info for <name of Product>" form.
  - When the "Tag Absent" value is selected, the tag record will be deleted from the "Tagged Data" form as well as from the corresponding field of the parent form after changes are saved.
- The *Value Type* field is used to determine the type of the tag value entered in the *Value Data* field:
  - "CheckBox" – in this case, it is not necessary to fill in the *Value Data* field
  - "String" – string value.
  - "Counter" – integer value (from "0" to "9").
  - "Tag" – the value of the tag must be either "Y" or "N".
  - "Money" – numeric value.
  - "Currency" – numeric currency code.
  - "Unknown" – no tag value type is specified, but the *Value Data* field must be filled in.
  - "List" – the tag value may be set as a list of values. In initial manual setup, a comma-delimited list of values is specified in the *Value Data* field (or in the *Custom Data* field of the higher-ranking form). After data is saved, each "List" type tag value will be displayed as a separate record in the "Tagged Data" form.

- The *Comment Text* field contains a description of the tag whose name is selected from the system list.
- The *Is Ready* field shows results of tag parameter check:
  - The field contains the "Ready" value if the check is successful.
  - The field contains the "Not Ready" value if errors have been detected during the check.

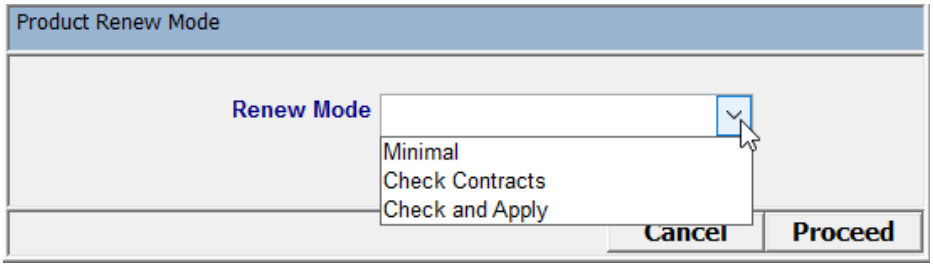
To check tag parameters for correctness, click the [Do...] button and select the "Check" value from the context menu. If an error is detected, a window with the corresponding message will open.

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

2.7 Approving Products and Applying Product Changes to Contracts

After adding a new Product, creating a Product by copying another Product's configurations or changing Product parameter values, the changes should be approved. To do so, click the [Approve] button in the "Products" form and select the "Approve" item in the context menu. When this item is selected, the "Product Renew Mode" form will open. In this form, select a mode for approving changes and click the [Proceed] button:

- "Minimal" – Product parameters are checked and registered in the database. The properties of contracts created earlier for this Product are not updated (and the correspondence of contract parameters to Product parameters is not checked).
- "Check Contracts" – in this mode, the Product's parameters are checked and registered in the database. A check is made that the parameters of contracts created for this Product match these parameters. Contract properties are not updated.
- "Check and Apply" – in this mode, the Product's parameters are checked and registered in the database. The parameters of contracts created for this Product are updated according to its new parameters.



"Product Renew Mode" form

If errors are found during checks, the appropriate message will be displayed.

After the Product has been successfully approved, an information window appears indicating that the changes have been made in the database. The *Is Ready* field value in the "Products" form becomes "Ready".

- If the parent Product does not require approval (the *Is Ready* field contains "Yes"), this subordinate Product and Products down the current branch of the hierarchy will be approved (the

parent Product's version and the version of other untouched subordinate lines of Products remain unchanged).

- If the parent Product must be approved (the *IsReady* field contains "No"), approval is called for the parent Product, and respectively, for the entire hierarchy (for all branches under the parent Product).

2.8 Applying Product Changes to Contracts

When a Product that was earlier used to register contracts is updated, the properties of the contracts and their accounts will be updated according to the value of the APPROVE_IMMEDIATE global parameter:

- When the parameter is set to "Yes" – immediately during Product approval.
- When the parameter is set to "No" – by clicking the [Approve] button and selecting "Apply" from the context menu in the "Products" form.
  When the value is "N", changes are applied in all cases within the Contracts Daily Update.
  For more information, see the document "Way4 Global Parameters".

> When approving contracts after changing an Account Scheme or Product, the correspondence of existing accounts to those of the new Account Scheme is checked. If they do not correspond, an error message is shown, the change is not approved and the contract remains in the original state (before the Product or Account Scheme was changed).

2.9 Creating Product Groups

In Way4, it is possible to group Products. Product groups are used during application import using the Advanced Applications module (see the document "Advanced Applications R2").

To create a database record for a new product group, select "Full → Configuration Setup → Products → Product Definition → Product Groups" from the user menu.

This will open the "Product Groups" form (see figure).

Grid for entering and editing Product group data

The form has the following fields:

- *Product Category* – select from a list of Product categories:

  - "Issuing" – issuing contract Products.
  - "Acquiring" – acquiring contract Products.
  - "Accounting" – bank system contract Products. This value is left for backward compatibility.
  - "Bank Accounting" – bank contract Products.
- *Name* – Product group name.
- *Code* – code used when importing external files into the system to link imported data with a set Product group.

Clicking the [Products] button opens the "Products for <name of group>" form used to set up Products from the group. The form is completely identical to the "Products" form (see the section "Main Product Parameters").

2.10 Configuring the "Contract Roles" List

The list of contract roles is configured in the "Contract Roles" form (Full → Configuration Setup → Products → Product Definition → Contract Roles), see figure.

Form for entering and editing contract role information

This form is used to configure additional classification of contracts in the contract hierarchy.

For issuing contracts, contract roles can be configured for main and supplementary cards, see the description of the Main Card Flag field. Roles such as "Company", "Store", "Device", and "Terminal" can be configured for acquiring.

This classification can be used in reports and to apply different tariffs (fees) to contracts with different roles (for example, different fees for the main and subordinate card).

This form contains the following fields:

- *Name* – role name.
- *Code* – record code.
- *Product Category* – selection from a list of Product categories.
- *Contract Category* – selection from a list of contract categories.
- *Main Card Flag* – indicates whether this is a main/supplementary card. Filled in for issuing card contracts and can be used in report generation. Values:
  - "Yes" – main card.
  - "No" – supplementary card.

A contract's role is set in the Product's *Contract Role* field (see the section "Additional Product Parameters") and is inherited for all contracts created on the basis of this Product. The

CONTRACT_ROLE parameter can be used to redefine the contract role can be redefined for a specific contract:

- In add_info_01/02/03/04 fields (when the parameter is set manually or with an application).
- In the ext_data field (when the parameter is set by an Event or with a custom procedure).

A role can be redefined on the contract level if one Product is used for contracts with different roles. For example, if a main card and supplementary card only differ by the fees that are charged (for example, an annual fee or issuing fee). In this case it is recommended to use one Product (for example, with the "Main Card" role) and define the supplementary card for the contract.

2.11 Default Contract Classifiers for a Product

The list of default contract classifiers for a Product is configured in the "Classifiers for <Product name>" form (Full → Configuration Setup → Products → Product Definition → Products → [Classifiers]).

Configuration of contract classifiers in a Product makes it possible to do the following:

- automatically assign classifier values to contracts when contracts are created (contracts created on the basis of the corresponding Product). This is done using classifiers with "Y" in the *Set To Contract* field of the "Classifiers for <Product name>" form.
- Use a default value set for a Product if this classifier is not set for the contract. This is done using classifiers with "N" in the *Set To Contract* field of the "Classifiers for <Product name>" form.

See the section "Configuring default classifiers" of the document "Way4 Client and Contract Classifiers".

2.12 "Configuration Groups" Classifiers

"Configuration Groups" classifiers are used to create custom views for database table data (e.g. to generate custom reports that use different terms), to group data, and to filter data by custom parameters.

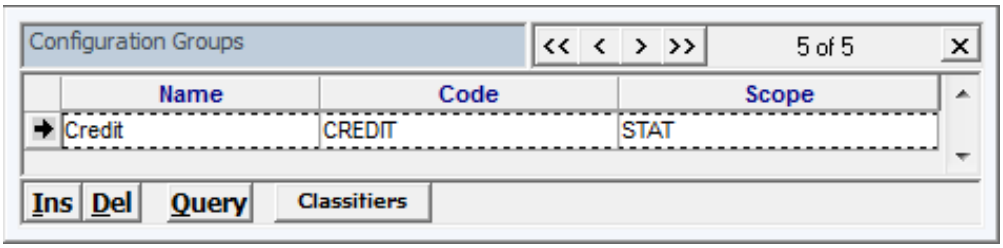
Further on, classified data can be used, for instance, to enter information to generate reports.

"Configuration Groups" classifiers can be used in Way4 to classify (mark) Products, contract subtypes, Events, and client and contract classifiers.

For example, several credit card Products are registered in the system for the Visa Classic card (Product 1, Product 2, etc.). At the same time, it is necessary to generate a report for the consolidated Visa Classic product. Using the configurations described in two sections that follow, the Products (Product 1, Product 2, etc.) will be marked with the same classifier value VISA CLASSIC and later on converted into a single product in a report. Therefore, these configurations allow for displaying consolidated parameters in a report.

2.12.1 Configuring "Configuration Groups" Classifiers

To set up "Configuration Groups" classifiers, select "Full → Configuration Setup → Client Classifiers → Configuration Groups" from the user menu. This will open the "Configuration Groups" form used to enter and edit information about classifier groups (see figure).

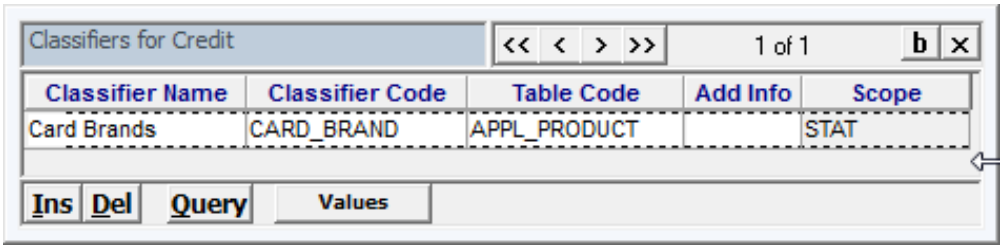


"Configuration Groups" form

To add a group of classifiers, add an empty row by clicking the [Ins] button and fill in the following fields:

- *Name* – name of the group
- *Code* – unique group code
- *Scope* – classifier's scope. A scope consolidates groups of classifiers, i.e. it is the highest level in a hierarchy of "Configuration Groups" classifiers.

Clicking the [Classifiers] button opens the "Classifiers for <name of group of classifiers>" form (see figure) used to set up classifiers.

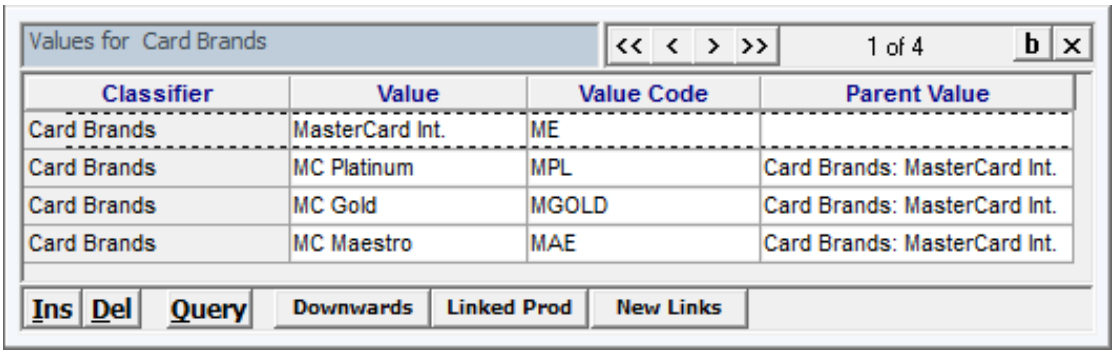


Form "Classifiers for <name of group of classifiers>"

To add a classifier, add an empty row by clicking the [Ins] button and fill in the following fields:

- *Classifier Name* – classifier name.
- *Classifier Code* – classifier code unique within this classifier's scope.
- *Table Code* – name of the table whose data is classified.

Clicking the [Values] button opens the "Values for <name of classifier>" form (see figure) used to set up classifier values.

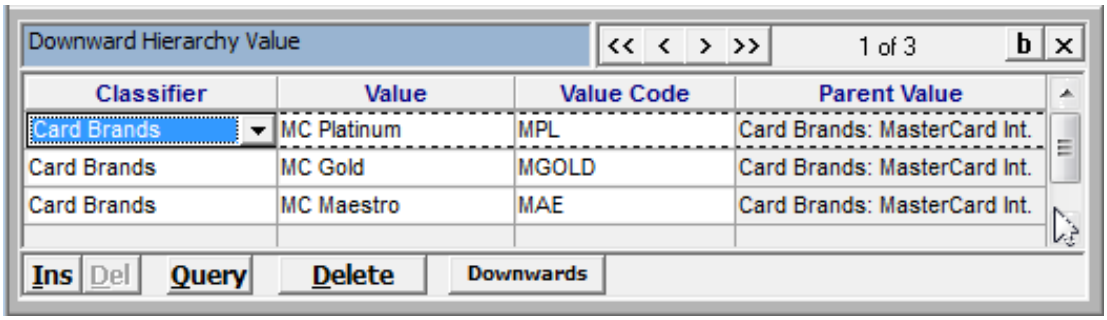


Form "Values for <name of classifier>"

To add a classifier value, add an empty row by clicking the [Ins] button and fill in the following fields:

- *Value* – classifier value.
- *Value Code* – code of the classifier value, unique within the classifier.
- *Parent Value* – this field is used to set up hierarchies of classifier values. In the field, select a value of this classifier or another classifier within the group that will be the parent value of the current value. A corresponding record will be automatically added in the "Downward Hierarchy Value" form of the parent value.

To access the "Downward Hierarchy Value" form (see figure), click the [Downwards] button.



Form "Downward Hierarchy Value"

This form is used to set up child values of the classifier value selected in the "Values for..." form. For this, add an empty row by clicking the [Ins] button and fill in the following fields:

- *Classifier* – drop-down list of classifiers from the group.
- *Value* – name of a child value from the specified classifier.
- *Value Code* – classifier value code.

When a record is added in the "Downward Hierarchy Value" form, a record with the corresponding parent value in the *Parent Value* field is automatically added in the "Values for..." form.

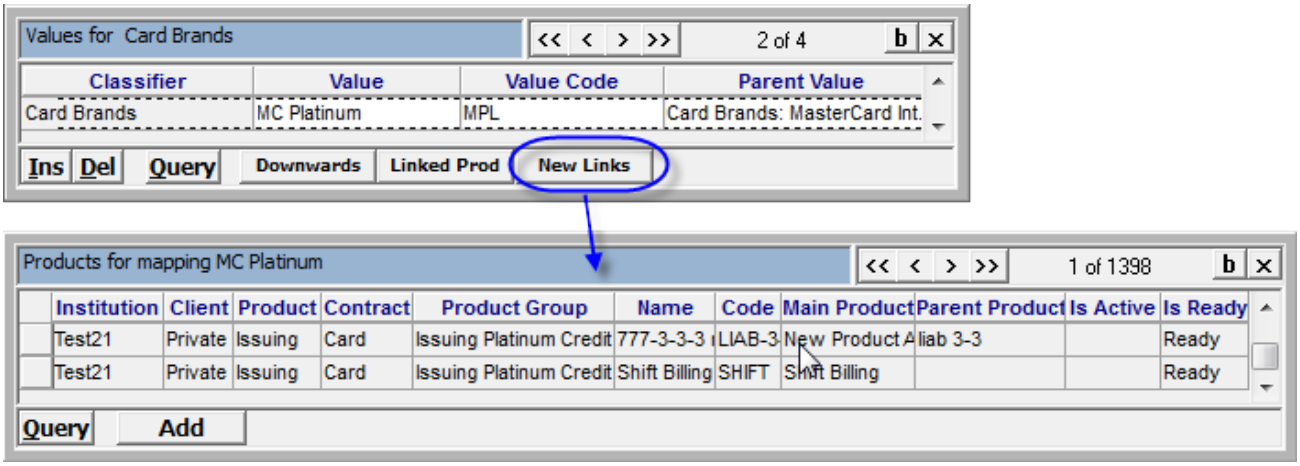
2.12.2 Marking Products Using "Configuration Groups" Classifiers

Data can be marked using "Configuration Groups" classifiers in the following ways:

- Data marking directly in the "Configuration Groups" form:
  - In the "Values for <classifier name>" form (Full → Configuration Setup → Client Classifiers → Configuration Groups → [Classifiers] → [Values], see figure), click the [New Links] button. A form will open that contains a list records of the table whose data is classified (the table set in the *Table Code* field in the "Classifiers" form, see the figure in the section "Configuring "Configuration Groups" Classifiers").
  - In the form that opens, select a record, click the [Add] button and execute the "Add One" command from the context menu. The record will be marked with the selected classifier value.
    If all table records must be marked with one classifier value, execute the "Add All" command from the context menu.

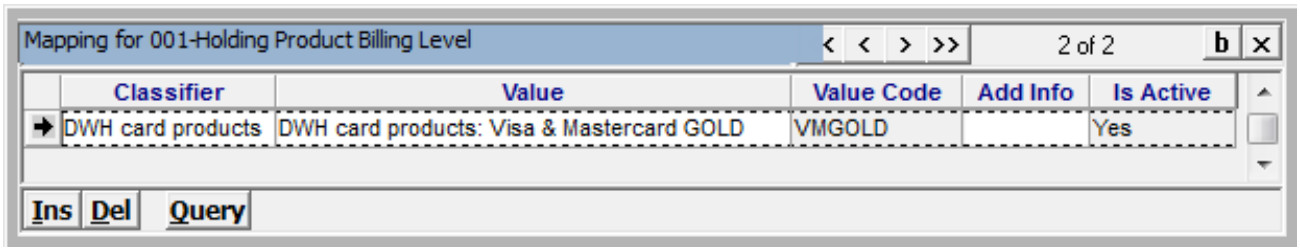
> A record marked by a classifier value will no longer be shown in the list.

- Clicking the [Linked Prod] button (see figure) opens a form containing table records marked by a selected classifier value. If marking must be deleted, click the [Delete] button and execute the "Delete Link" command from the context menu to delete the selected record or "Delete All Links" to delete all marking.



Marking data with "Configuration Groups" classifiers

- Marking data in the Products form:
  - To mark data in the form of additional Product parameters (Full → Configuration Setup → Products → Product Definition → Products → [Full Info]), click the [Mapping] button.
  - In the "Mapping for <Product name>" form that opens (see figure), click the [Ins] button and fill in the fields:
    - In the *Classifier* field, select a classifier from the list of registered classifiers.
    - In the *Value* field, select the classifier value.
    - The *Value Code* field is filled in automatically when a value is selected in the *Value* field.

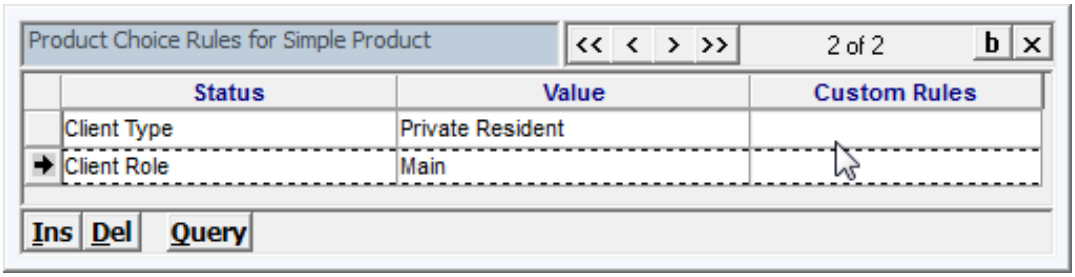


Form "Mapping for <Product name>"

2.13 Configuring Available Products

The list of Products that are available to a client is determined by classifiers values assigned to the client (or client properties).

Rules for selecting Products are set in Product properties in the "Product Choice Rules" form (Full → Configuration Setup → Products → Product Definition → Products → [Full Info] → [Choice Rules]), see figure.



"Product Choice Rules" form

This form contains the following fields:

- *Status* – drop-down list to select the client classifier name.
- *Value* – field for selecting the value of the classifier specified in the *Status* field for which the Product will be available to the client.
- *Custom Rules* – field for entering additional parameters.

For more information about client classifiers, see the document "Way4 Client and Contract Classifiers".

2.14 Changing a Product when Reissuing a Card

If it is necessary to change the Product for a card when reissuing the card (scheduled reissue, reissue after loss), do as follows:

- The old and new Product must be configured for one main Product, in the corresponding hierarchy. I.e. a card Product in a "Main/Sub" hierarchy can be replaced with a Product in the "Main/Sub" hierarchy (a Product in a "Main/Sub" hierarchy cannot be replaced with a Product in a "Liability" hierarchy).
- Set the REPLACE_TO=<code of new Product> tag in the *Custom Data* field of the "old" card Product.
- For the "old" card Product, it is recommended to set "Closed" in the *Is Active* field. This setting doesn't prevent cards issued for the Product from working and does not affect processing of the REPLACE_TO=<Product code> tag. I.e. reissued ("old") cards will work as usual, but cards will be reissued with the new Product. This setup is optional.
- If the aforementioned settings have been made, a new card contract will be created when reissuing a card.
  Reissue is initiated during daily procedures (Contracts – Daily Update) and when the following menu items are run "Issuing → Mark/Unmark Card To Production → Mark/Unmark Single Card for Plastic Replacement", "Issuing → Mark/Unmark Card To Production → Mark/Unmark for Plastic Replacement by Selection", "Issuing → Mark/Unmark Card for Production → Mark/Unmark Lost/Stolen Card".

> Note that when a card is marked, "Produce Card" must be selected in the *Card Event* field of the "Production Type" form, and, accordingly, the Event related to creation of a card contract must be selected in the Event field. See the section "Card Issuing" of the document "Issuing Module".

When reissuing a card with a new Product, the following must be considered:

- If a card being reissued has related cards, the REPLACE_TO tag should be set in the main card's Product and in the related card's Product.
- If classifiers were set in the old card contract and these classifiers are not set in the new Product, classifiers will be copied from the old card to the new card contract. Classifiers set in the new Product are inherited to the card contract in the usual way.
- Settings like credit limit, individual payment orders, individual tariffs, and contract addresses that were made in the old card contract are copied to the new card contract (i.e. actions are performed similar to those when reissuing a card without changing the Product).
- Usage limiter values are moved to the new contract when the following conditions are met simultaneously:
  - If there is a limiter with the same code in the new Product's Service Package.
  - An individual limiter with this code was active for the old contract before reissue.

2.15 Changing the Product Line for Main and Supplementary Cards

Cards with different roles can be created for an account contract: main cards and supplementary cards. See the section "Role Setup and Mapping Products with Contract Roles".

The card Products used for these cards can belong to different Product lines. This is done by mapping Products with a classifier that has the PRODUCT_LINE code. A Product line is a set of Products mapped with one PRODUCT_LINE classifier value.

If a new main card in a different Product line is created for an account contract, new contracts corresponding to the new Product line can be automatically created for supplementary cards of the original main card.

This feature makes it possible, for example, to issue supplementary cards when a client is issued a Visa Platinum card to replace a current Visa Gold card (with a set of supplementary cards).

New contracts are created only for supplementary cards that are already open for the same account contract.

Card Products must be correctly mapped with contract roles and Product lines (PRODUCT_LINE classifier).

Setup procedure:

- Roles are set up and Products are mapped with roles, see the section "Role Setup and Mapping Products with Contract Roles".
- A classifier with the PRODUCT_LINE code is configured, see the section "Setup of a Classifier with the PRODUCT_LINE Code".
- A Product line is set up (card Products are mapped with a PRODUCT_LINE classifier), see the section "Product Line Setup (Mapping Products with a PRODUCT_LINE Classifier)".
- Synchronization is enabled (mode for automatic creation of new supplementary card contracts), see the section "Enabling Synchronization Mode".
- The synchronization procedure (automatic creation of new supplementary card contracts) is described in the section "Synchronization".

2.15.1 Role Setup and Mapping Products with Contract Roles

Contract roles are configured in the "Contract Role" form.

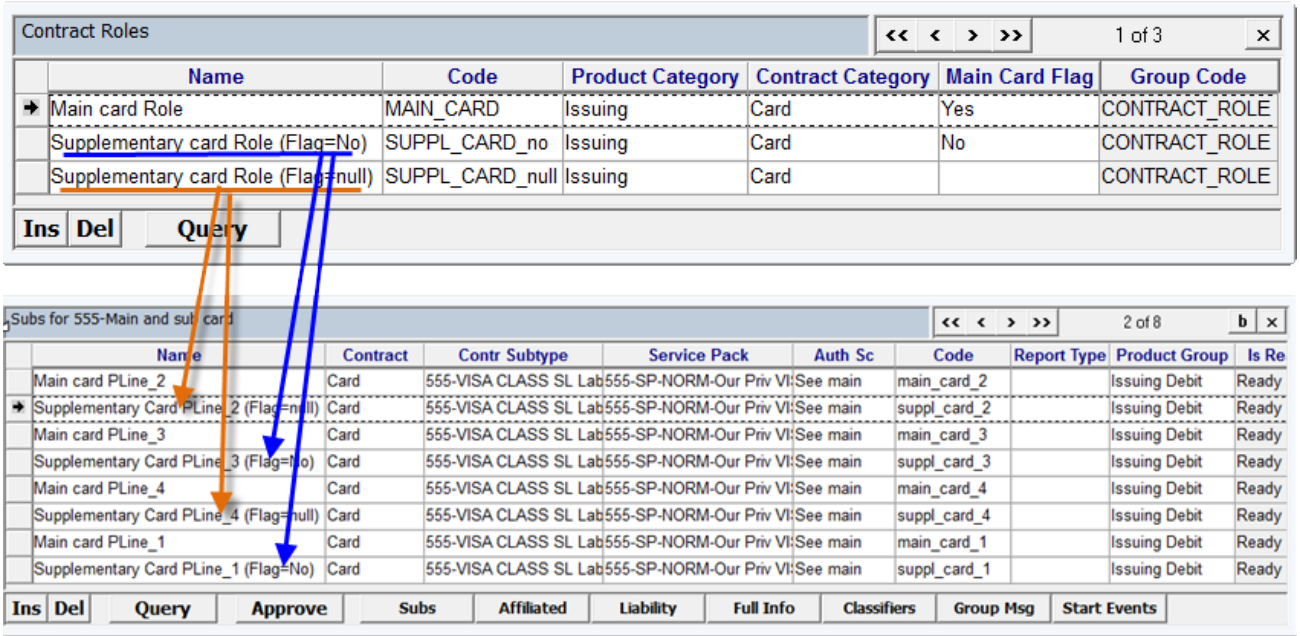
The value in the *Main Card Flag* field of the "Contract Role" form indicates whether a card is a main or supplementary card:

- "Yes" – main card.
- "No" – supplementary card.
- The *Main Card Flag* field does not have to be filled in; in this case, the card is also considered to be supplementary and a separate Product in a Product line can be configured for it.

For a detailed description of contract role setup, see the section "Configuring the "Contract Roles" List".

A configured role for a contract is set in a Product's *Contract Role* field (see the section "Additional Product Parameters") and is inherited for all contracts created on the basis of this Product.

An example of configuration (mapping) of card contracts for one account contract with consideration of roles and Product lines is shown in figure below figure.



Configuration of card contracts for one account contract with consideration of roles and Product lines

2.15.2 Setup of a Classifier with the PRODUCT_LINE Code

A hardcoded PRODUCT_LINE classifier is set up in the "User Classifier" form, see figure.

User Classifiers — 1 of 1

Name	Code	Group Code	Category	Default Value	Applies To	Client Cat	Contract Cat	Product Cat	Role	Add Info	On Off Mode	Domain Code
PRODUCT_LINE	PRODUCT_LINE		Classifier		Product				Primary			

Ins | Del | Query | Validate | Values | Mapping

"User Classifier" form

Specifics of classifier setup:

- In the *Code* field, set the PRODUCT_LINE value.
- In the *Applies To* field, set the "Product" value.
- In the *Role* field, set the "Primary" value.

Configure possible values for the classifier, see the example in figure below.

Values for PRODUCT_LINE — 1 of 4

Weight Factor	Name	Code	Result Event Code	Is OK Value	Add Info	Is Active	Date From	Date To
0	PRODUCT_LINE_1	PRODUCT_LINE_1		Yes		Yes	00/00/0000	00/00/0000
0	PRODUCT_LINE_2	PRODUCT_LINE_2		Yes		Yes	00/00/0000	00/00/0000
0	PRODUCT_LINE_3	PRODUCT_LINE_3		Yes		Yes	00/00/0000	00/00/0000
0	PRODUCT_LINE_4	PRODUCT_LINE_4		Yes		Yes	00/00/0000	00/00/0000

Ins | Del | Query

"Values..." form

2.15.3 Product Line Setup (Mapping Products with a PRODUCT_LINE Classifier)

A product line may consist of one or several Products for the main card and one or several Products for supplementary cards. However, for a certain role for supplementary cards with a certain *Main Card Flag* value (either "N" or not filled in), only one Product in one Product line can be set up.

Products for a main card and supplementary cards in the same line must be mapped with the same classifier value (see figure).

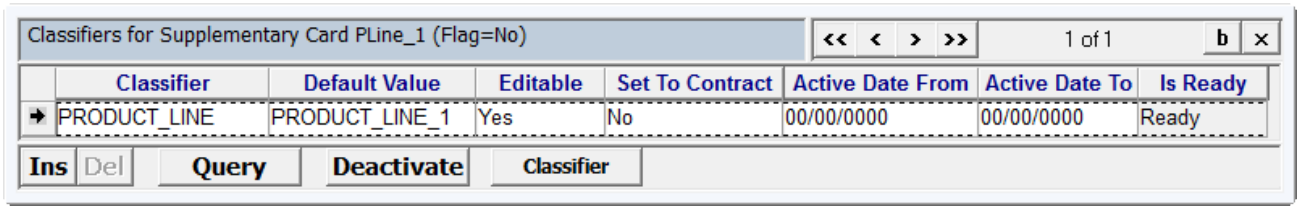
A classifier value is assigned to a Product in the "Classifiers for <Product name>" form (see figure).

Classifiers for Main card PLine_1 — 1 of 1

Classifier	Default Value	Editable	Set To Contract	Active Date From	Active Date To	Is Ready
PRODUCT_LINE	PRODUCT_LINE_1	Yes	No	00/00/0000	00/00/0000	Ready

Ins | Del | Query | Deactivate | Classifier

Mapping a main card's Product



Mapping a supplementary card's Product

Specifics for filling in the form's fields:

- In the *Classifier* field, a classifier with the PRODUCT_LINE code is specified; in the *Default Value* field, select a value for mapping this Product line.
- In the *Set To Contract* field, specify "N".

For more information about classifier setup in Products, see the section "Default Contract Classifiers for a Product".

2.15.4 Selecting Cards for Reissue (Decisions Setup)

This setup is used when:

- There are several active main cards and one must be selected for reissue. By default (i.e. without this setup), when there are several active main cards, **there is no reissue** and an error message is generated.

  > Active cards are those with the "Active" or "Not Ready" value in the *Approval* (IS_READY) field of a card contract's form.

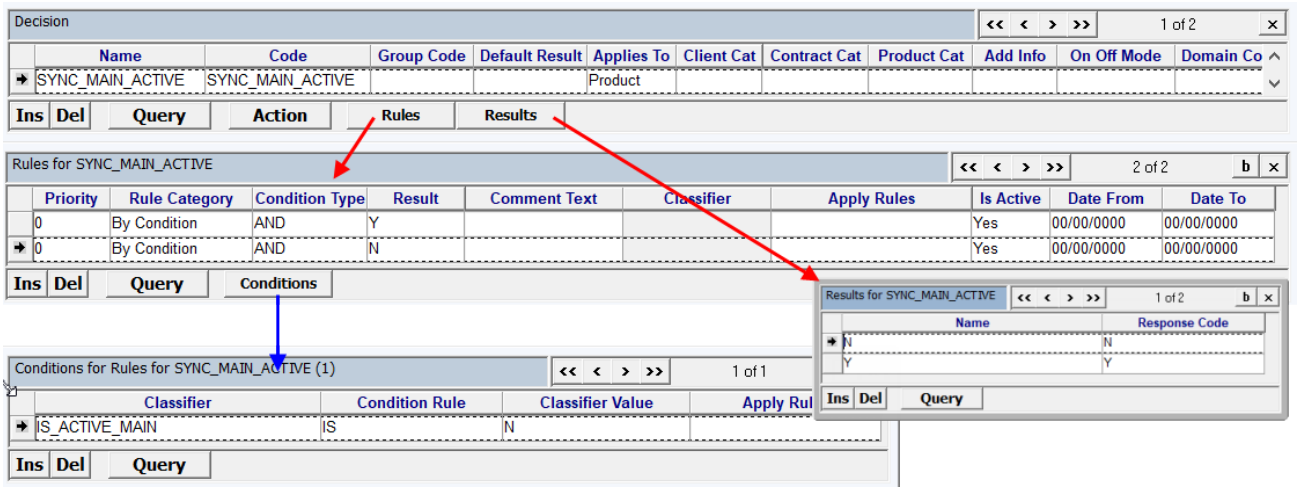
- Select a Supplementary card for reissue. Without this setting, all Supplementary cards are reissued.

Setup procedure:

- Configure a calculated classifier/classifiers (Decision):
  - A Decision for selecting the main card from several active cards for reissue.
    For example, if there are two active Main cards, one with the "Decline" status and the other with the "Valid" status, a Decision can be used to configure the rule for selecting the main card with the "good" status (i.e. with the "Valid" status).
  - A Decision for selecting Supplementary cards for reissue. A Decision can be used to set the rule according to which cards with the "bad" status (with a particular classifier value) are not reissued.

  > One Decision can be configured for selecting the main card and Supplementary cards.

A Decision must have two values: "Y" (this value must be returned if the card is considered active) and "N" (this value must be returned if the card is considered inactive).
A Decision can be configured on the basis of a standard classifier (CONTR_STATUS) or on the basis of a custom classifier used to map card contracts.
For more information about Decision setup, see the section "Calculated Classifiers (Decisions)" of the document "Way4 Client and Contract Classifiers".
An example of Decision setup is shown in figure.

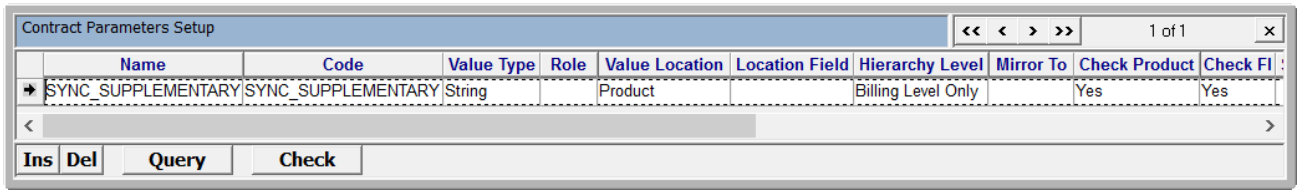


Example of Decision setup for selecting a main card for reissue

- Specify the code of the configured classifier (classifiers) as the value of the global parameter:
  - SYNC_MAIN_ACTIVE_DECISION – to specify a decision for selecting the main card.
  - SYNC_SUPPL_ACTIVE_DECISION – to specify a decision for selecting supplementary cards.

2.15.5 Enabling Synchronization Mode

To enable the mode for synchronization in the account contract's Product, the contract parameter SYNC_SUPPLEMENTARY=Y must be set.



Form "Contract Parameters Setup"

The following settings are recommended for the custom parameter:

- In the *Value Location* field, specify "Product".
- In the *Hierarchy Level* field, specify "Billing Level Only".

With these settings, the parameter is specified in an account Product. The parameter can be set using a tag or a classifier (this is determined by the parameter's settings; for more information about contract parameter setup, see the section "Contract and Client Custom Parameters" of the document "Way4 Client and Contract Classifiers").

The parameter's value can be set for a financial institution. In this case, "Yes" must be specified in the *Check FI* field.

2.15.6 Synchronization

When approving a card contract with a Product with the "Main" role for the first time, the following actions are taken when SYNC_SUPPLEMENTARY=Y is present:

- Based on mapping with the PRODUCT_LINE classifier, the Product line for the card is determined (i.e. the value that is set for the classifier).
- For the account contract, a search is made for a current active card that has the role with the "Main" value (**with which synchronization has not yet been performed**) and its Product line is determined. If the Product line for the new and current card does not differ (i.e. the Product of the new Main card is mapped with the same classifier value), no actions will be taken with supplementary cards. A contract is created for one main card with a new Product (a contract is not automatically created for a current supplementary card).
- However, a situation is possible when a main card and supplementary card are in the same Product line (for example, line 1), a main card is created in another line (for example, line 2), after which a main card is created again in line 1. In this case, the supplementary card in line 1 will be created automatically, since the "previous" main card belongs to another Product line.

  > An account contract can have only one active Main card in a certain (in one) Product line. An account contract can have several active Main cards in different Product lines.

- If the Product line of a current (most recently issued main card; that has been synchronized) main card and new main card differ, all supplementary active cards in the same line as the main card for which synchronization has not yet been performed are checked:
  - In the new Product line, a search is made for the Product of a supplementary card with the role that matches that of the original supplementary card (including the *Main Card Flag* value).
  - If a suitable Product is found, a contract is created. If there are several original supplementary cards with the same role, new contracts will be created for each card.
    If a new card should not be created for a supplementary card:
    - The supplementary card's Product line must differ from that of the main card or must not be specified (this means that the supplementary card's Product is not mapped with a PRODUCT_LINE classifier). In this case, a new card contract is not created.

    Or

    - When Product lines differ, a new card is not created if for the new contract in the new Product line, a Product with a role that matches the supplementary card's role is not found (including matching of the *Main Card Flag* value). I.e. if a Product with the same role is found.

- When supplementary cards are created, the main card's production Event is inherited from the new main card to the supplementary cards (PRODUCTION_EVENT field of the CARD_INFO table).
- For a new card, a hardcoded SYNC_SUPPLEMENTARY Event is called. This Event is used to transfer additional data from the original contract to the new contract. This is done using custom procedures. For more information, contact OpenWay.

2.16 Redefining Contracts and Contract Accounts

Various Product settings – Service Package, Account Scheme, Tariff settings, etc. are used to redefine a fee/interest contract and contract account.

2.16.1 Redefining a Fee Contract and Contract Account

2.16.1.1 General Information

Fee contracts/accounts may need to be redefined, for example, in the following cases:

- If for different tariff plans a fee must be recorded in different bank contracts.
- If fees for different transaction types must be recorded separately.
- If fees must be charged from or credited to contracts that are not involved in the transaction but have a relation (for example, contractual) to this transaction.

A fee contract/account can be redefined in one of the following ways:

- The fee contract and fee account to which a fee is credited can be redefined using the FEE_CONTRACT and REDEF_FEE_ACC_CODE tags. This is possible for main Services and for custom fee Services.
- The contract and account from which a fee is charged can be redefined with the CONTRACT_FOR and REDEF_ACC_CODE tags. This is only possible for custom fee Services.

These tags can be set for a Service, or for the tariff that is specified in the *Fee Tariff* field of the Service.

Specifics of redefinition:

- When tags are set in a tariff with the "Service" role (tariff specified in the Service's *Fee Tariff* field), the contract and account that are being redefined can be set for the "Service" role tariff itself, for an independent tariff, or in a calculated classifier's properties:
    - When specifying a redefined contract and account in a tariff with the "Service" role, note the following:
        - When the fee contract and fee account to which a fee is credited is redefined for the main Service, the FEE_CONTRACT tag and REDEF_FEE_ACC_CODE=<account type code> tags are used.

- When redefining for a custom fee, the tags CONTRACT_FOR, FEE_CONTRACT and REDEF_ACC_CODE=<account type code>/REDEF_FEE_ACC_CODE=<account type code> are used.
- The tags REDEF_ACC_CODE, REDEF_FEE_ACC_CODE, FEE_CONTRACT, CONTRACT_FOR are specified in the tariff's *Apply Rules* field.
- The tags REDEF_ACC_CODE=<account type code>/REDEF_FEE_ACC_CODE=<account type code> can be used, for example, together with the ANY_CONTR_TYPE tag when one Product can be used for different contract subtypes. In this case, the tags make it possible to split recording the fee by different accounts depending on contract subtype.

- When a redefined contract and account are specified in an independent tariff (a tariff that differs from the one specified in the Service) or in a calculated classifier:
  - The tags FEE_CONTRACT and REDEF_FEE_ACC_CODE are used with FROM_DECISION, FROM_TARIFF constructions.
  - The tags FEE_CONTRACT and REDEF_FEE_ACC_CODE are used with FROM_DECISION, FROM_TARIFF constructions.
  - The tags CONTRACT_FOR and REDEF_ACC_CODE are used with FROM_DECISION, FROM_TARIFF constructions.
  - When the tag REDEF_FEE_ACC_CODE=FROM_TARIFF:<tariff type code>; is used, a search is made for a tariff in the contract for crediting the fee from the Service's *Fee Contract* field, or the contract can be redefined using the FEE_CONTRACT tag.
  - When the tag REDEF_ACC_CODE=FROM_TARIFF:<tariff type code>; is used, a search for the tariff is made in the contract for charging the fee. This is a Source or Target contract accordingly, which can be redefined using the CONTRACT_FOR tag.
  - It is not recommended to use the tags REDEF_FEE_ACC_CODE=FROM_TARIFF:<tariff type code>; or REDEF_ACC_CODE=FROM_TARIFF:<tariff type code>; to redefine the code of the account for crediting/charging a fee. It is recommended to use a FROM_DECISION construction to do so.
  - The tags REDEF_ACC_CODE, REDEF_FEE_ACC_CODE, FEE_CONTRACT, CONTRACT_FOR are specified in the Service's *Service Details* field.

These settings make it possible to analyse contract/client states, classifiers, custom parameters, contract hierarchy, document parameters (including tags), and to select a fee contract/account depending on specific conditions.

2.16.1.2 Usage Guidelines

Recommendations for using the FROM_DECISION and FROM_TARIFF values:

- It is recommended to use a FROM_DECISION value for the following:
  - To configure simple conditions for selecting an account (contract). For example, depending on the value of a contract classifier or on the value of a document tag (for example, selection of a fee account depending on the transaction type).

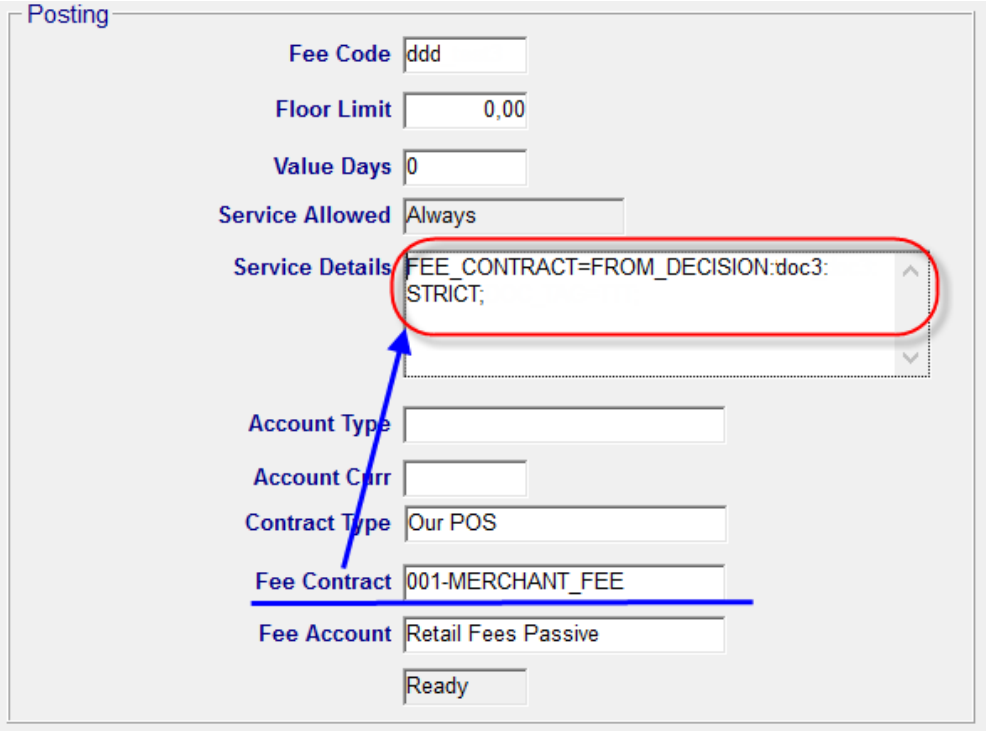
- It is recommended to use a FROM_DECISION value to redefine a fee account. A contract can be redefined using a FROM_DECISION value, but tariffs are better suited to address this task.

- It is recommended to use the FROM_TARIFF value in the following cases:
  - If selection of a fee contract depends on Product settings – on the Product that is used, tariff plan, Service Package, etc.
  - To optimize Product settings, a FROM_TARIFF construction makes it possible to reduce the number of Services configured.
    For example, one Service with the FROM_TARIFF setting can be created, according to which a fee will be recorded in different accounts depending on the transaction type.
  - For complex analysis of Product settings, preferred counterparties, document checks, related to limits and other checks that are made when searching for a tariff.
  - It is recommended to use the FROM_TARIFF construction for redefining a fee contract. It is not recommended to use tariffs for redefining accounts (to do so, the FROM_DECISION construction is recommended).

2.16.1.3 Using Decisions for Redefinition

The use of decisions to redefine fee contracts and accounts is set up according to the following scheme:

- In the appropriate Services, set the FEE_CONTRACT=FROM_DECISION:<decision code>; tag (redefinition of the contract to which the fee is transferred), and/or CONTRACT_FOR=FROM_DECISION:<decision code>; tag (redefinition of the contract from which the fee is charged).
  Example of redefining the bank contract from the *Fee Contract* field for a custom fee, see figure – when a transaction is made using this Service, the fee contract (001-MERCHANT_FEE) can be redefined using a decision with the code "doc3".

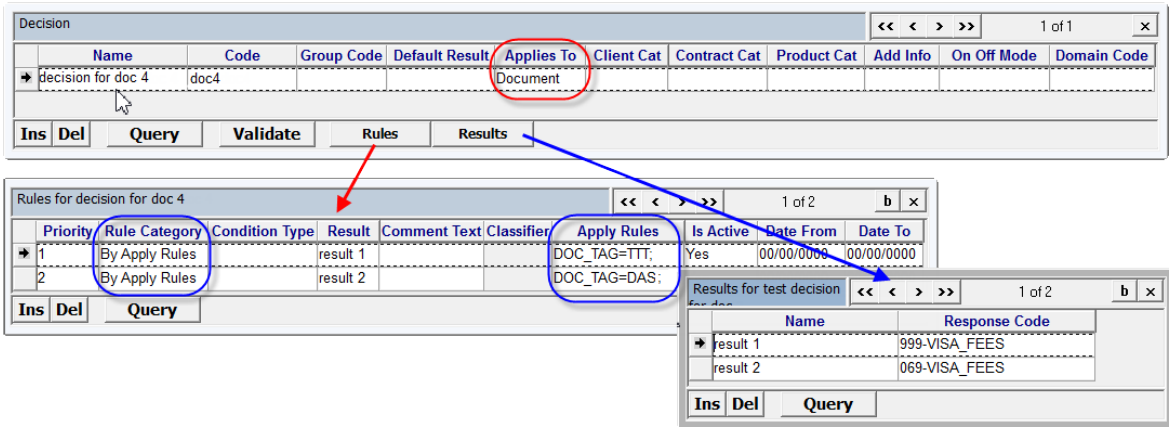


Configuring a custom fee to redefine the fee contract

- Configure a decision:
  - In the classifier's *Apply To* field, select "Document".
  - In the *Rule Category* field of the rule ("Rules for <classifier name>" form) select "By Apply Rules".
  - The contract number is specified in the *Response Code* field of the decision's "Results for <classifier name>" form. This number will be used to record a fee if the conditions in the *Apply Rules* field of the "Rules for <classifier name>" form are met.

    > By default, this contract number is used as a template. That is, the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). For this contract number to be used as it is specified in the "Results for..." form, use the ":STRICT" postfix in the value of the FEE_CONTRACT/CONTRACT_FOR tags. For example, FEE_CONTRACT=FROM_DECISION:REV_CONTR_ISS:STRICT;

  - The conditions under which this contract will be used should be configured using tagged parameters in the *Apply Rules* field of the "Rules for <classifier name >" form. Tags for checking document parameters can be specified in this field (for example, DOC_TAG/DOC_TAG_VALUE, SOURCE_CHANNEL, TARGET_CHANNEL, IF_CURRENCY, etc.), and/or tags that check contract parameters (for example, IF_CS, IF_PARM group tags), see figure.



Configuring a decision for redefining a fee contract

For more information about configuring decisions, see the section "Calculated Classifiers (Decisions)" of the document "Way4 Client and Contract Classifiers".

- The tags REDEF_ACC_CODE=FROM_DECISION:<decision code>;, REDEF_FEE_ACC_CODE=FROM_DECISION:<decision code>; must be used to redefine a fee account. Redefinition is performed in the same way as for a contract. The account type code is always used as it is specified in the *Response Code* field of the "Results for <classifier name>" form (i.e. the ":STRICT" postfix is not used).

2.16.1.4 Using Tariffs for Redefinition

A tariff with the "Redefinition" role is used to redefine a fee contract and account:

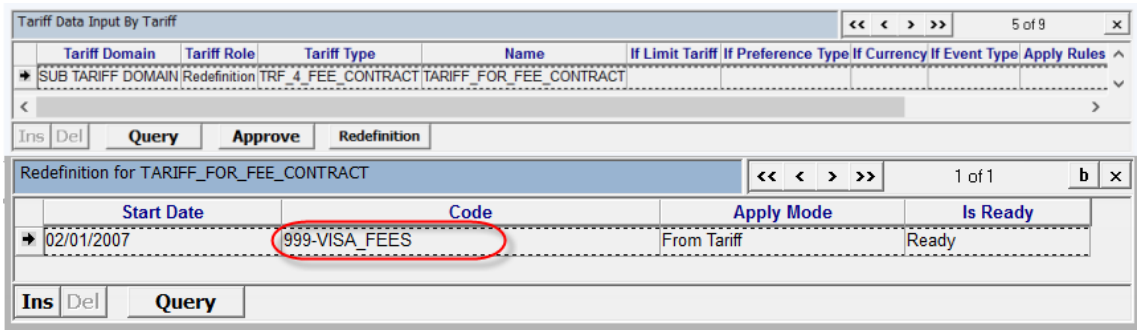
- In the appropriate Services, set the tag FEE_CONTRACT=FROM_TARIFF:<tariff type code>; and/or CONTRACT_FOR= FROM_TARIFF:<tariff type code>. When a transaction is made with this Service, the contract to which the fee is transferred/from which the fee is charged can be redefined using the tariff with this code. In the example in figure, the bank contract from the *Fee Contract* field (001-MERCHANT_FEE) can be redefined using a tariff with the TRF_4_FEE_CONTRACT code.

Configuring a Service (Custom Fee) to redefine a fee contract

- Configure a tariff with the "Redefinition" role:
  - The contract number is specified in the tariff's *Code* field. This number will be used to record a fee if this tariff is selected.

    > By default, this contract number is used as a template. That is, the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). For this contract number to be used as it is specified in the *Code* field, use the ":STRICT" postfix in the FEE_CONTRACT/CONTRACT_FOR tags. For example, FEE_CONTRACT=FROM_TARIFF:REV_CONTR_ISS:STRICT;.

  - If conditions for selecting a tariff must be configured, this is done normally. For more information about tariff setup, see the document "WAY4 Advanced Tariff Management". Tags used to check document parameters (DOC_TAG/DOC_TAG_VALUE, SOURCE_CHANNEL, TARGET_CHANNEL, IF_CURRENCY, etc.), and/or tags for checking contract parameters (for example, IF_CS, IF_PARM tags) can be specified in the tariff's *Apply Rules* field, see figure.



Configuring a tariff to redefine a fee contract, without analysing document tags

It is not mandatory to configure conditions for selecting a tariff (checking document parameters, etc.). I.e. a contract can be redefined simply when it uses a specific tariff plan, without additional conditions (see figure).

Configuring a tariff to redefine a fee contract, with analysis of document tags

- Tariffs can also be used to redefine fee accounts, but this setup is not recommended. The FROM_DECISION construction should be used if it is necessary to redefine a fee account (see the section "Using Decisions for Redefinition").

2.16.2 Redefining an Interest Contract

2.16.2.1 General Information

It may be necessary to redefine interest contracts, for example, if interest for different tariff plans must be recorded in different bank accounts.

An interest contract (i.e. an account template's *Interest Contract* field) is redefined using the INTEREST_CONTRACT=FROM_DECISION:<decision code>; tag or INTEREST_CONTRACT=FROM_TARIFF:<tariff type> tag in an account template or Account Scheme. I.e. decisions or tariffs are used for redefinition.

These settings make it possible to analyse contract/client states, classifiers, custom parameters, contract hierarchy, and to select an interest contract depending on specific conditions.

2.16.2.2 Usage Guidelines

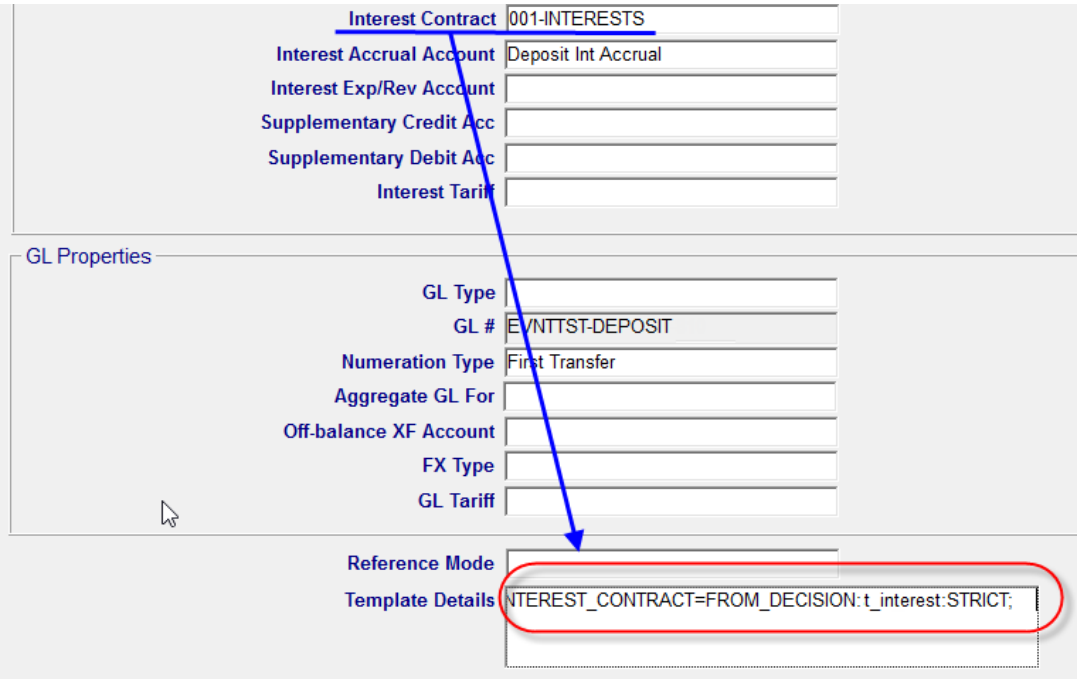
Recommendations for using the FROM_DECISION and FROM_TARIFF values:

- It is recommended to use the FROM_DECISION value to configure simple conditions for selecting a contract. For example, depending on a contract classifier's value.
- It is recommended to use the FROM_TARIFF value in the following cases:
  - If selection of an interest contract depends on Product settings – on the Product that is used, tariff plan, Service Package, etc.
  - To optimize Product settings, the FROM_TARIFF construction makes it possible to reduce the number of account templates that are configured.

2.16.2.3 Using Decisions for Redefinition

Use of decisions to redefine an interest contract is configured according to the following scheme:

- In the Account Scheme or account template, set the INTEREST_CONTRACT=FROM_DECISION:<decision code>; tag (see figure). The interest contract will be determined based on a decision with this code and on the appropriate contract parameters that are checked according to conditions set in the classifier.



Configuring redefinition of an interest contract using a decision

> The INTEREST_CONTRACT tag value in an account template has higher priority.

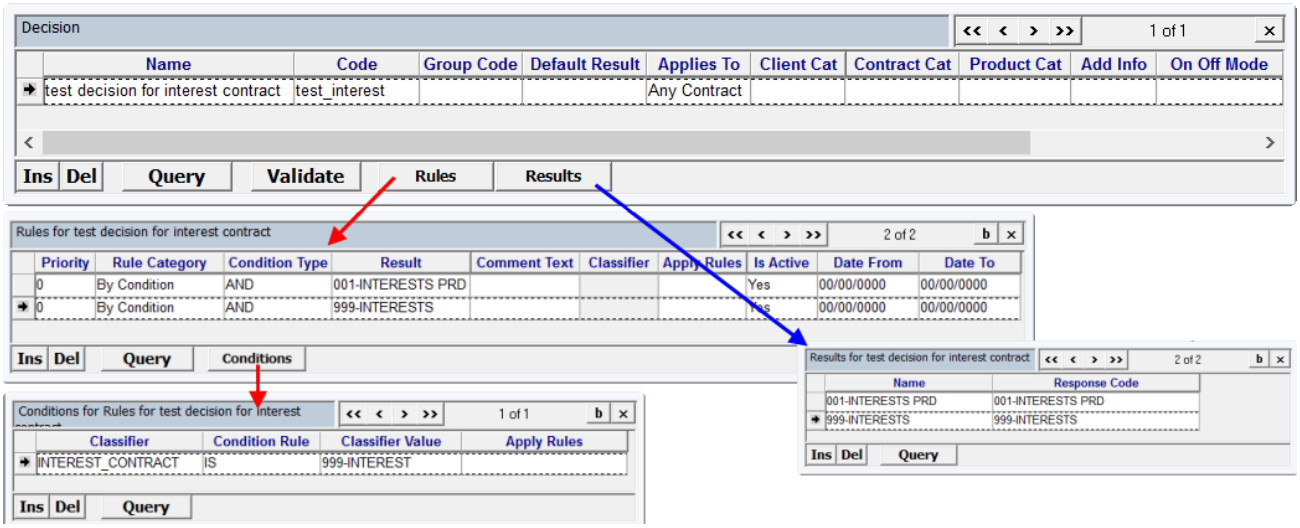
The INTEREST_CONTRACT tag value set in an Account Scheme works for all scheme templates. If a particular template has its own INTEREST_CONTRACT tag, its value will be used.

- Configure a decision:
  - Decisions are configured normally. See the section "Calculated Classifiers (Decisions)" of the document "Way4 Client and Contract Classifiers".

- A bank contract number in the *Response Code* field of the decision's "Results for <classifier name>" form, see figure.

> By default, the bank contract number specified in this field is used as a template. That is, the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). For this contract number to be used as it is specified in the "Results for..." form, use the ":STRICT" postfix in the value of the INTEREST_CONTRACT tag. For example, INTEREST_CONTRACT=FROM_DECISION:REV_CONTR_ISS:STRICT;

- A decision's value (i.e. the interest contract's number) is determined on the basis of these contract/client classifiers (in the example in figure below depending on the value of the INTEREST_CONTRACT contract classifier).



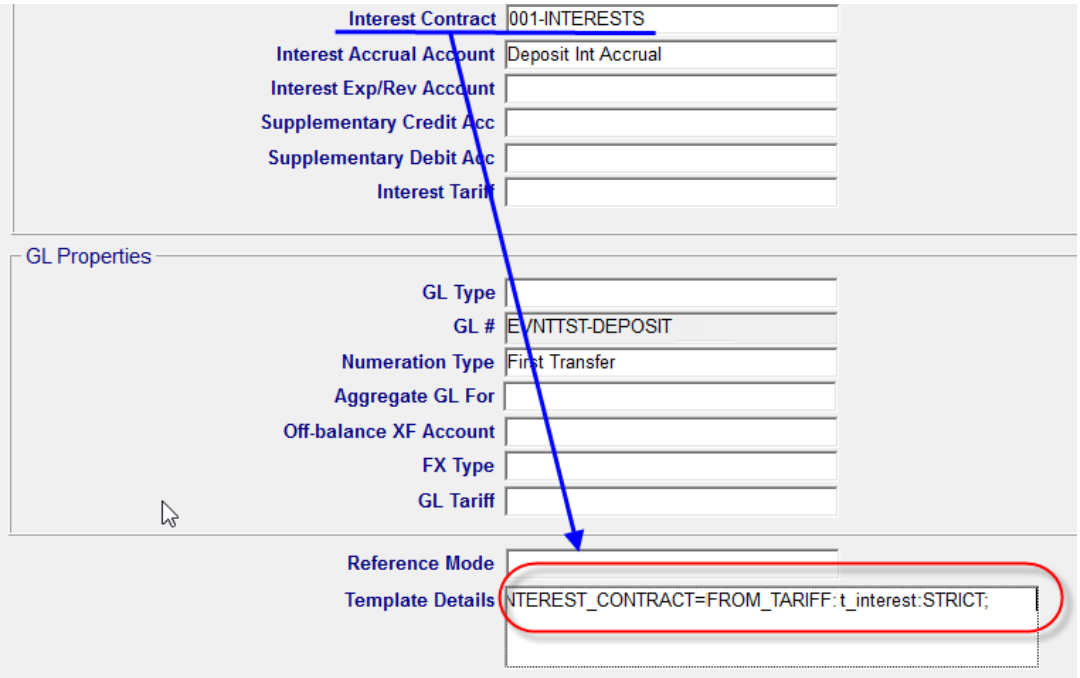
Configuring a decision for redefinition of an interest contract

For more information about configuring decisions, see the section "Calculated Classifiers (Decisions)" of the document "Way4 Client and Contract Classifiers".

2.16.2.4 Using Tariffs for Redefinition

Redefinition of an interest contract using a tariff with the "Redefinition" role is configured according to the following scheme:

- In the Account Scheme or account template, set the tag INTEREST_CONTRACT=FROM_TARIFF:<"Redefinition" tariff type code>;. An interest contract will be determined based on a tariff with this code (in the example in figure below – with the "t_interest" tariff type code).



Configuring redefinition of an interest contract using a tariff decision

> The INTEREST_CONTRACT tag value in an account template has higher priority.

The INTEREST_CONTRACT tag value set in an Account Scheme works for all scheme templates. If a particular template has its own INTEREST_CONTRACT tag, its value will be used.

- Configure a tariff with the "Redefinition" role:
  - The bank contract number is specified in the tariff's *Code* field.

    > By default, the bank contract number specified in this field is used as a template. That is, the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). For this contract number to be used as it is specified in the *Code* field, use the ":STRICT" postfix in the INTEREST_CONTRACT tag. For example, INTEREST_CONTRACT=FROM_TARIFF:t_interest:STRICT; (see figure).

  - If conditions for selecting a tariff must be configured, this is done normally. For more information about tariff setup, see the document "WAY4 Advanced Tariff Management".

  It is not mandatory to configure conditions for selecting a tariff (checking document parameters, etc.). I.e. a contract can be redefined simply when it uses a specific tariff plan, without additional conditions (see figure).

Configuring redefinition of an interest contract using a tariff decision

3 Contract Types and Subtypes

A contract type is a classifier used to group contract subtypes. A contract type is also used to specify default values of some of contract subtype parameters.

A contract subtype determines a set of contract parameters, such as message channel, numeration method, etc., for a certain financial institution and client category.

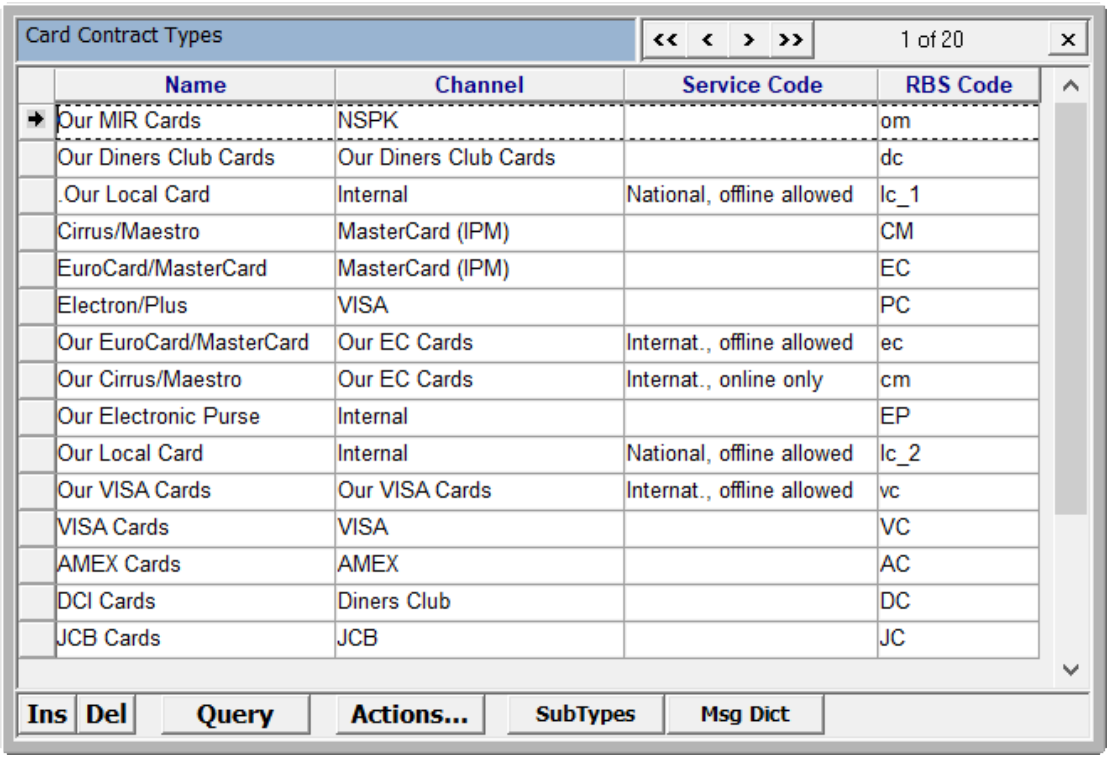
3.1 Contract Types

Way4 provides a standard set of contracts that define the way a bank works with a payment system.

To create a contract type, select the user menu path "Full → Configuration Setup → Contract Types":

- Card contract types are created in the "Card Contract Types" form.
- Device contract types are created in the "Device Contract Types" form.
- Issuing/acquiring contract types are created in the "Accounting Contract Types" form.

The figure shows the "Card Contract Types" grid used for creating card contract types.



Card contract types grid

The "Card Contract Types" grid contains the following fields:

- *Name* – name of contract type.
- *Channel* – processing channel corresponding to this contract type.

- *Service Code* – name of the service code for this contract type according to the affiliated payment system. The field's list of values is generated using the handbook "Card Service Codes")Full → Configuration Setup → Contract Types → Card Service Codes).
- *RBS Code* – field is used to link uploaded data with this contract subtype when external files are received by the system.

The "Device Contract Types" and "Accounting Contract Types" forms contain the same field sets with the following differences:

- Neither form contains the *Service Code* field.
- The "Device Contract Types" form, used for creating device contracts, contains an additional field *Terminal Category* in which the device type should be indicated ("ATM"/"POS"/"Imprinter"/"Infokiosk").

The context menu command "Validate" of the [Actions] button is used for the following checks:

- To check the *RBS Code* and *Name* fields for uniqueness in the contract type, client type, and financial institution.
- To check the fields of all subordinate subtypes, to fill in the *BIN Record* and *IPS Product* fields of card contract subtypes (see the description of the *BIN Record* and *IPS Product* fields in the section "Checking Contract Subtype Fields").
- To check the uniqueness of the *Name* field for applet subtypes in a certain card contract subtype (Full → Configuration Setup → Contract Types → Card Contract Types → [Applets]).

The "Copy Type" context command of the [Actions] button is used to copy the contract type. During this procedure, the system will create a new contract type and automatically create its contract types with the same properties as those of the copied contract (see the section "Entering Contract Subtype Data"). After copying, it is necessary to set up the following objects for the new contract type:

- Transaction subtypes (see the section "Transaction Subtypes" of the document "Documents").
- New Services in the existing Service Package (if necessary).
- Create Service Packages for a new contract type (see the document "Way4 Service Packages").

The [SubTypes] button is used to create and configure contract subtypes (see the section "Entering Contract Subtype Data").

The [Msg Dict] button is used to access the "Msg Dict for <name of contract type>" form, containing a record from the "Message Dictionary" dictionary (Full → Configuration Setup → Main Tables → Message Dictionary) corresponding to the selected contract type. In the form that opens, names of contract types specified in the *Message Name* field can be translated into national languages registered in the corresponding system dictionary (Full → Configuration Setup → Client Classifiers → Languages). For more information, see the section "Message Dictionary" of the document "Way4 Dictionaries".

3.2 Entering Contract Subtype Data

Several contract subtypes can be created for each contract type. They serve as parameters when creating contracts in the database. When creating a card contract, a contract subtype can be specified for a limited range of card numbers. In this way, the card number serves as an information source on

the card. For instance, various ranges of card numbers indicating the respective Bank Branch, card project etc. can be created within a single BIN range designating a payment system product.

To create a new contract subtype, follow these steps:

- Open the contract type form, selecting the user menu path "Full → Configuration Setup → Contract Types → <Client Category> Contract Types" (see an example in the figure in section "Contract Types").
- Select the desired type among those already defined.
- To display data for all the existing contract subtypes of the selected contract type, click [SubTypes] in the "<Contract Type> Contract Types" form.
- Click [Ins] to add a blank record in the "SubTypes for <Contract type>" form (see the figure in section "Card Contract Subtypes Form") and fill in all the record fields one by one.

3.2.1 Card Contract Subtypes Form

The figure shows the form "SubTypes for <name of card contract type>".

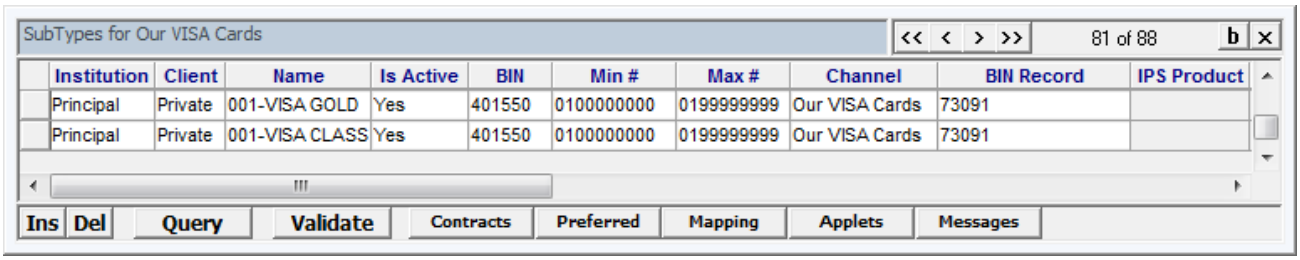


Table of subtypes for selected card contract types

The form contains the following buttons:

- The [Contracts] button is used to access the list of contracts registered in the system using this subtype.
- The [Preferred] button is used to configure preferred counterparties (see the section "Configuring preferred counterparties (Preferred)").
- The [Mapping] button is used to mark contract subtypes using "Configuration Groups" classifiers. For more information, see the section "Default Contract Classifiers for a Product".
- The [Applets] button is used to set up contract subtypes for additional card applications (Applets).
- The [Validate] button is used to check subtype fields and to fill in the *BIN Record* and *IPS Product* fields of the contract subtype (see the sections "BIN Record", "IPS Product", "Checking Contract Subtype Fields").

The card contract subtype grid (see figure) contains the following fields:

3.2.1.1 Institution

The *Institution* field indicates the financial institution to which the contract subtype is affiliated.

3.2.1.2 Client

The *Client* field determines the client category, where "Private" is an individual, "Commercial" is a legal person, and "Accountant" is a bank branch.

3.2.1.3 Name

The *Name* field is the contract subtype name; the following name format is recommended: "NNNname", where NNN are the last three digits of the Financial Institution number registered in the Way4 database (see the value in the *Branch Code* field in the form that is opened using the menu item "Full → Configuration Setup → Main Tables → Financial Institutions"). This field's value must be unique within the contract type, client type and financial institution.

Note that subtypes with the same code (see the *RBS Code* field) in different financial institutions must have the same names following the financial institution prefix. I.e. subtypes with the same code must have the same meaning in all financial institutions.

Subtype names are checked when checking subtype fields (see the section "Checking Contract Subtype Fields").

3.2.1.4 Is Active

The *Is Active* field sets the subtype state: "Yes" – stands for active, "No" – inactive.

While working with contracts, operators may need to stop using a certain contract subtype, e.g. when a card project is closed or if the bank suffers losses caused by fraudulent cards issued in a card number range belonging to the subtype. The inactive status of a subtype does not prevent cardholders from using existing contracts. However, the plastic cannot be issued/reissued.

If an inactive subtype is set for an active Product (see descriptions of the *Date From*, *Date To*, and *Is Active* fields for a Product in the section "Additional Product Parameters"), when an attempt is made to create a contract with this Product, an error message is generated, and the contract is created with the "Not Ready" status. To prohibit creating new contracts with inactive subtypes, mark card Products with these subtypes as inactive in the *Date From*, *Date To*, and *Is Active* fields for the Product (see the section "Additional Product Parameters").

When "Replace Only" is specified in this field, a card can be reissued for an existing contract with this subtype. A new contract with this subtype cannot be created.

3.2.1.5 BIN

The *BIN* field is used to create BINs affiliated with a payment system. This value is used as a prefix for automatically generated contract numbers.

> For correct processing of authorizations and clearing, it is recommended to set contract subtypes for the entire BIN Range issued by the payment system to the bank or processing center.

3.2.1.6 Min#, Max#

Fields *Min#* and *Max#* are respectively the lower and upper limits of the card number range specified for contracts of the created subtype. Card contract numbers are generated randomly from within the range of the BIN indicated earlier.

3.2.1.7 Channel

The *Channel* field is the message channel corresponding to the subtype. When you create a new contract subtype, it inherits the Message Channel of the contract type.

3.2.1.8 BIN Record

The *BIN Record* field is filled in when the [Validate] button is clicked in the form of a certain subtype, when the [Actions] button's "Validate" context menu command is executed in the contract type form, or when the menu item "Full → Configuration Setup → Contract Types → Contract Types Validation" is run. A search is made for a record in the BIN table corresponding to this subtype. A search in the BIN table is made according to a range of card numbers (*Start BIN*, *End BIN* fields) and the *Sub BIN* field. In the search, the number of matched characters is progressively reduced to the first three characters in the *Sub BIN* field. First, a search is made for six characters (total match), and if a record is found, the search ends. If no record is found, a search is made for five characters, etc.

> When searching for a record, if other conditions are the same, a BIN table record with "VISA" in the *Brand* (CARD_BRAND) field has a higher priority than a record with a different value in this field.

According to the results of searching for a record in the BIN table, the following messages may be displayed:

- "Corresponding BIN Table Record not found" – the *BIN Record* field was not filled in when the [Validate] button was clicked, since no record was found in the BIN table.
- "Several Corresponding BIN Table Records found. Manual validation is recommended" – this message is displayed if several records meeting the search conditions were found in the BIN table. In this case:
  - A random record for filling in the *BIN Record* field is selected from the list found.
  - In the drop-down list of the *BIN Record* field, records from the BIN table with the corresponding *Sub BIN* are shown (number range is not considered).
  - In this case, it is recommended to manually select the required record in the *BIN Record* field list.
- "Selected BIN Table Record is absent" – the *BIN Record* field was filled in earlier, when the [Validate] button was clicked, the corresponding record in the BIN table was not found.
- "Contract Subtype successfully validated" – a unique record was found in the BIN table, the *BIN Record* field was successfully filled in.

Results of the last search for a record in the BIN table can be viewed using the [Messages] button of the form "SubTypes for <card contract type name>". If no record is found, an "Error" message is generated for active subtypes, or a "Warning" message for inactive subtypes.

In the contract subtype's *BIN Record* field, the values of the following fields from the "BIN Table" form are specified in the following format: <*Sub Bin* field value>:<*Product ID* field value, i.e. IPS product name>:<*Usage* field value>.

In the following cases the contract subtype's *BIN Record* field is automatically filled in:

- When approving a Product with a subtype with a specific channel for which in the "Message Channels" form (Full → Configuration Setup → Main Tables → Message Channels), the *Contra Channel* field is filled in and "No" is specified in the *Is On Us* field.
- When checking a financial institution (see the section "Checking FI Parameters" of the document "Financial Institutions").
  The global parameter IGNORE_IPS_PRODUCT affects filling in the *BIN Record* field (for more information, see the section "Checking Contract Subtype Fields").
  If the *BIN Record* field is filled in, records are checked (see the section "Checking Contract Subtype Fields").

3.2.1.9 IPS Product

The *IPS Product* field specifies the name of the card product in the payment system.

The *IPS Product* field is filled in when the [Validate] button is clicked in the form of a certain subtype, when the [Actions] button's "Validate" context menu command is executed in the contract type form, or when the menu item "Full → Configuration Setup → Contract Types → Contract Types Validation" is run (see the description of the BIN Record field). The field is filled in with the value of the *Product Name* field from the "IPS Product Definition" form (see the section "Payment System Product Dictionary" of the document "Way4 Dictionaries").

A contract subtype's *IPS Product* field is filled in automatically when a Product is approved, when checking a financial institution (for more information, see the section "Checking Contract Subtype Fields").

The global parameter IGNORE_IPS_PRODUCT affects filling in the *IPS Product* field (for more information, see the section "Checking Contract Subtype Fields").

3.2.1.10 Expire For New

The *Expire For New* field is the default validity period of a new card specified in months.

3.2.1.11 Expire For Renew

The *Expire For Renew* field is the default validity period of a replaced card if the card contract is renewed; the period is specified in months.

3.2.1.12 Service Code

The *Service Code* field contains the name of the service code for cards of this subtype, specified by the payment system. This field's value is usually the same as the *Service Code* field value of the corresponding Contract Type (see the figure in the section "Contract Types"). The form's list of values is generated using the handbook "Card Service Codes" (Full → Configuration Setup → Contract Types → Card Service Codes).

3.2.1.13 PM Code

Through this field, Card Production properties within a BIN can be selected for the created subtype. Specify the *PM Code* field value equal to the *Code* field value of the "Parameters for ..." form that is opened from the "Bank Production Parameters" form (Full → Configuration Setup → Card Production Setup → Bank Production Parameters). For more information about card production parameters, see the section "Card Production Parameters" of the document "Configuring WAY4 for Magnetic Stripe Card Issuing".

3.2.1.14 Numeration Type

Allows numeration rules to be set for a card contract:

- "Custom" – an individual part of the contract number is generated according to a custom procedure.
- "Random" – an individual part of the number is randomly selected from a possible range (see the *Min #* and *Max #* fields). This is the default value.
  When setting a range, note that one digit (the final one) in the value of the *Min#* and *Max#* fields is not used to generate a unique card number (PAN). In the PAN this final digit is used as a check bit (its value is calculated based on the PAN's other digits). I.e. the specified range will actually be smaller by one digit.
- "Sequential" – an individual part of the contract number is generated sequentially in increasing order by adding digits to the number indicated in the *Current #* field.
- "Manual" or any other value differing from those listed above – manual entry of a number.

> It is not recommended to use the "Sequential" or "Random" values if a contact number can be set manually or is defined by external systems.

3.2.1.15 Validation Type

This field is used to specify special parameters in the form of tags. For more details, see the section "Tags used when configuring Products and contract subtypes" of the document "Setup Tags".

If assigning several of the above-mentioned parameters to a contract subtype, the parameters must be differentiated by a semicolon placed between them.

3.2.1.16 Plastic Code

The plastic type code that is used to produce plastics for the contract subtype is specified in the *Plastic Code* field. The field's list of values is generated using the handbook "Plastic Codes" (Full → Configuration Setup → Contract Types → Plastic Codes). See the section Configuring the "Plastic Codes" Handbook".

The plastic type code can be set in the card contract using the contract custom parameter PLASTIC_CODE.

3.2.1.17 RBS Code

The *RBS Code* field is used to link loaded data with the defined contract subtype when external files are received by the system. This field's value must be unique within the contract type, client type and financial institution.

The *RBS Code* field can be set in the following formats:

- <financial institution prefix with a value between 3 and 6 characters>-<subtype code>.
- The *RBS Code* field's value can be set in another format, possibly without numeric values. For example, "ec-ck".

> If *RBS Code* is set as <financial institution prefix with a value between 3 and 6 characters>-<subtype code>, when copying a Product to another financial institution, the contract subtype's *RBS Code* field is updated according to the new prefix (according to the prefix of the new financial institution). If the *RBS Code* field does not contain a value in digits, the field will not be updated. In this case, it is assumed that the *RBS Code* field value is set in a different format and does not contain a financial institution prefix.

Note that subtypes with the same code but in different financial institutions must have the same name after the financial institution's prefix in the *Name* field. I.e. subtypes with the same code must have the same meaning in all financial institutions.

Subtype names are checked when checking subtype fields (see the section "Checking Contract Subtype Fields").

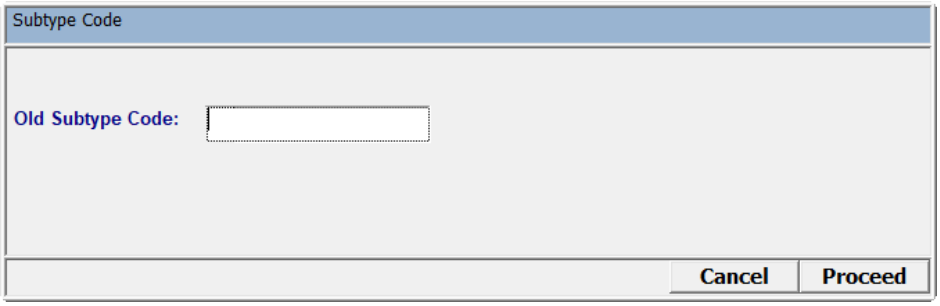
If one subtype code has several different names (without the financial institution prefix), correct these names. When the standard classifier CONTR_SUBTYPE is used and there are several different names for subtypes with the same code, the name of the classifier's corresponding value will be arbitrarily selected. This means that the name of the classifier's value will be correctly displayed in the interface for one financial institution's contracts only, and incorrectly for other financial institutions.

After correcting the names of subtypes with the same code in different financial institutions, synchronize the changes to subtypes with the CONTR_SUBTYPE classifier. To do this, use the menu item "Full → DB Administrator Utilities → Special OpenWay Utilities → Product Utilities → Update contract subtypes classifier".

If the value of the *RBS Code* changes, the changed value must be synchronized with the corresponding value of the standard classifier with the CONTR_SUBTYPE code. To do so, click the [Actions] button and run the context menu command "Synchronize Classifier". In the "Subtype Code" form that opens,

specify the subtype's old code and click the [Proceed] button. The value of the CONTR_SUBTYPE classifier for contracts with this subtype will change to the new value.

Note that use of the old RBS code in IF_CS constructions is not checked; in these constructions the old code should be manually changed to the new one.



Synchronization of a changed RBS code with the CONTR_SUBTYPE classifier

3.2.1.18 Add Parms

This field is used to specify special parameters of contract subtypes in the form of tags. For more details, see the section "Tags used when configuring Products and contract subtypes" of the document "Setup Tags".

3.2.1.19 Chip Scheme

Smart card parameter scheme that will be assigned by default to contracts with this subtype.

This parameter redefines the value of the *Chip Scheme* field on the Service Package level.

3.2.2 Account and Device Contract Subtype Forms

The grids for account and device contract subtypes are similar to the card contract subtype grid (see the figure in section "Card Contract Subtypes Form") with the following differences:

First, these forms do not contain fields typical of card contract subtypes (for example, *BIN*, *Plastic Code*).

Also, several fields have been added to the form for configuring contract numeration. Account contract numbers and device contract numbers can be entered manually or automatically created during contract approval, depending on how the values of the fields described in the sections "Prefix", "Num Type", and "Current #".

The [Validate] button in this form is used to check that subtype fields are filled in (see the section "Checking Contract Subtype Fields").

3.2.2.1 Prefix

This value is used as a prefix for automatically generated contract numbers.

3.2.2.2 Num Type

This field indicates the contract numeration type:

- "Custom" – an individual part of the contract number is generated according to a custom procedure.
- "Manual" – the number needs to be entered manually when the contract is created.
- "Random" – an individual part of the number is randomly selected from a possible range (see the *Min #* and *Max #* fields).

> By default, if "Random" is set, a check digit is not used when numbering account contracts. To use a check digit, set the value of the global parameter GENERATE_LUHN to "Y".

- "Sequential" – an individual part of the contract number is generated sequentially in increasing order by adding digits to the number indicated in the *Current #* field.

3.2.2.3 Current

This field is used for storing the last number generated for a contract during automatic contract numeration. If a value is manually entered into this field, all contracts created afterwards will receive increasing numbers from that value. The *Current#* field value will also be updated automatically.

3.3 Checking Contract Subtype Fields

Card contract, account contract, and device contract subtype fields are checked according to the following rules:

- Values in the *Min#* and *Max#* fields must have the same number of digits, the value of the *Max#* field must be greater than that of the *Min#* field.
- *Min#, Max#, Prefix* field values may not contain spaces.
- The total length of values in the *Prefix* and *Min#* fields may not exceed 24 digits.

When checking card contract subtype fields, the *BIN Record* and *IPS Product* fields are filled in.

If the fields are not filled in, they will be filled in (see the sections "BIN Record" and "IPS Product").

If the *BIN Record* field is filled in, the record is checked:

- A check is made for this record in the BIN table and it is compared with other table records in the appropriate range.
- If the record assigned earlier has the "Inactive" or "Closed" status in the BIN table and another record with the "Active" status is found in the appropriate range and it has the same *Data Source* field value, the subtype is assigned the new active record from the BIN table.
- If the record assigned earlier has the "Inactive" or "Closed" status in the BIN table and another record with the "Active" status is found in the appropriate range, but with a different value in the *Data Source* field, the record assigned to the subtype is not changed.

> An exception is when "Own" is set in the *Data Source* field for an old inactive (or closed) record, and the new record has a value other than "Own" in this field. In this case, the contract subtype will be assigned the new record.

When checking/comparing records, if all other conditions are the same, a BIN table record with "VISA" in the *Brand* (card_brand) field has a higher priority than a record with another value in this field.

A check is made of the *Name* field for subtypes with the same code in different financial institutions (see the description of the *RBS Code* field), and a check that the subtype name matches the name of the classifier value:

- A check is made that for one subtype code (RBS Code field), all active subtypes have the same name in the *Name* field (without the prefix that is the financial institution code).
- A check is made that the subtype name matches the name of the corresponding CONTR_SUBTYPE classifier value (if it is used).

Methods for checking:

- The [Validate] button in the contract subtype form (see the sections "Card Contract Subtypes Form" and "Account and Device Contract Subtype Forms") checks the fields of a certain contract subtype.
- Using the [Actions] button's "Validate" context menu command in the contract type form (see the section "Contract Types") checks the fields of all a certain contract type's subtypes.
- When a Product is approved.
- When checking a financial institution (see the section "Checking FI Parameters" of the document "Financial Institutions").
- When a contract is approved. The *BIN Record* and *IPS Product* fields are not filled in.
  The check is made for both active and inactive subtypes (see the description of the *Is Active* field).
  If it is necessary to disable checking contract subtype fields (including disabling filling in the *BIN Record* and *IPS Product* fields), use the IGNORE_IPS_PRODUCT global parameter (see the document "Way4 Global Parameters").
  - To disable the check for all financial institutions, set the value of the global parameter IGNORE_IPS_PRODUCT to "ALL".
  - To disable the check for a specific institution, set the tag IGNORE_IPS_PRODUCT=ALL; in the institution's *Special Parms* field.
  - To disable the check for a specific subtype, set the tag IGNORE_IPS_PRODUCT; in the subtype's *Add Parms* field.

  When the IGNORE_IPS_PRODUCT parameter is set (i.e. when validation of subtype fields is disabled), a Product will be approved (and an institution validate) when the *BIN Record* and *IPS Product* fields are not filled in or filled in incorrectly. A "Warning" message will be generated (i.e. an message about an error that it is not mandatory to fix).

Note that when the IGNORE_IPS_PRODUCT parameter is set (when the *BIN Record* and *IPS Product* fields are not filled in or filled in incorrectly), Way4 Datamart ETL processes (processes for loading data to Way4 Datamart), "QMR" and "QOC" reports, and statistics reports for the Russian Central Bank (for example, form 0409250 reports) will work incorrectly.

3.4 Configuring preferred counterparties (Preferred)

Users may set additional conditions for operations in the Way4 system according to preferred counterparty parameters.

The form "Preferred for <name of subtype>" can be accessed by clicking the [Preferred] button in the form "SubTypes for <name of card contract type>" (see the figure in section "Card Contract Subtypes Form"). For more information about working with preferred counterparties, see the document "Preferred Counterparties".

3.5 Configuring Message Templates (Group Msg)

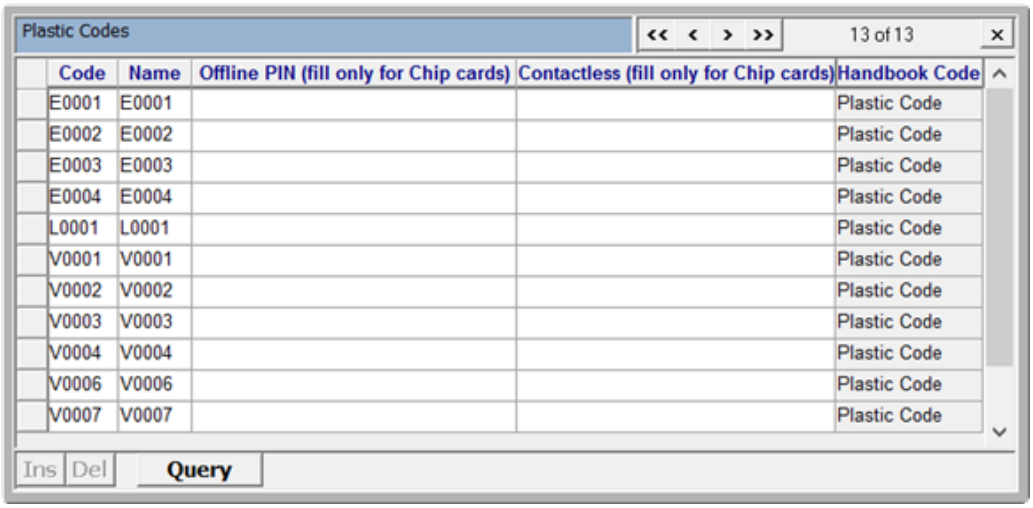
Way4 allows information or marketing messages to be sent to a client whose contract uses the given Product.

To configure message templates, click the [Group Msg] button in the "Products" form (Full → Configuration Setup → Products → Product Definition → Products"), see the figure in section "Main Product Parameters". For more information about setting up message templates, see the section "Configuring Message Templates" of the document "Configuration of Client Messages".

3.6 Configuring the "Plastic Codes" Handbook

The list of *Plastic Code* field values for a card contract subtype (see the section "Plastic Code") is generated using the "Plastic Codes" handbook (Full → Configuration Setup → Contract Types → Plastic Codes), see figure.

The handbook is used when issuing bankcards and in generating reports.



"Plastic Codes" form

This form contains the following fields:

- *Code* – plastic type code.
- *Name* – plastic type name.
- *Offline PIN (fill only for Chip cards)* – ability to enter a PIN offline. Possible values: "Yes"/"No". Only filled in for chip cards.
- *Contactless (fill only for Chip cards)* – indicates a contactless card. Only filled in for chip cards. Possible values:
  - "Yes" – contactless cards.
  - "Combined" – combined cards.
  - "Micro Tags" – tokens.
  - "No" – contact cards.

3.7 Default Contract Classifiers for Contract Subtypes

A default classifier configured for a contract subtype is used if this classifier is not set for the contract.

Default contract classifiers for a contract subtype are configured as tags in the *Add Parms* field of the "Sub Types" form (Full → Configuration Setup → Contract Types → <Client Category> Contract Types → [Sub Type]).

See the section "Configuring default classifiers" of the document "Way4 Client and Contract Classifiers".

3.8 Working with contract subtypes

3.8.1 Registering new subtypes

To use a BIN range for card production, it is necessary to register a special contract subtype for this range for each financial institution (Full → Configuration Setup → Contract Types → Card Contract Types → [SubTypes]). Form fields are described in the section "Entering Contract Subtype Data".

> When registering a subtype, the value in the *BIN* field must match the current IIN in the international payment system.

3.8.2 Seamless reissue of card contracts in a new subtype with the same Product

Way4 supports the ability to gradually reissue cards that are about to expire, in a new range. Procedure:

- For the old contract subtype, specify RANGE_CLEANUP; in the *Add Parms* field. This tag indicates a subtype that must be gradually freed up. Card contracts that haven't expired can still be used.
- Create a new subtype (see the section "Registering new subtypes").
- For all Products that have the current (old) subtype, set the new subtype.
  If a script is used to replace a Product's subtype, the "Apply Product Changes" procedure is not required.

In scheduled reissue of a card whose subtype is marked with the RANGE_CLEANUP; tag, at the marking stage, a search is made for the subtype in the Product and the card is reissued by creating a new contract with a new number within the new subtype that is specified for the Product.

> This approach can be used, for example, in freeing up unused BIN ranges when changing to 8-digit BIN numbers. For more information, contact OpenWay.

3.8.3 Seamless reissue of card contracts in a new subtype with a different Product

If a card's Product must be changed in scheduled reissue of a card and a change must be made to a new range specified in the properties of the new Product's subtype, use the tag REPLACE_TO=<new Product's code > for the old Product. For more information, see the section "Changing a Product when Reissuing a Card".

Card contracts that haven't expired can still be used.

3.8.4 Range expansion

For Products for which a large number of cards will be issued, we recommend combining several consecutive ranges in advance and gradually introducing them, when the preceding range is full.
Procedure:

Ranges are expanded (or combined) using a contract subtype's *Max#* field.

For example, when there are three BIN ranges (see below), they can be combined by creating one subtype with the value "50...9" in the *Min#* field and "53....0" in the *Max#* field:

Min#	Max#
50...9	51...0
51...9	52...0
52...9	53....0

3.8.5 Closing a range for further issue of new cards

If it is necessary to close an old range so that new cards are no longer issued in this range, and allow only reissue of existing cards, set the value "Replace Only" in the *Is Active* field for the old subtype (see the section "Card Contract Subtypes Form").

3.8.6 Issuing cards according to applications

When an application to issue a card contract is received, a subtype is selected according to the subtype code directly indicated in the application or according to the subtype that is specified Product settings (the Product code is specified in the application).

> ⚠ A subtype that is marked with the RANGE_CLEANUP tag cannot be specified for a Product, since this makes it impossible to issue and reissue cards.

When processing the application, a check is made that the subtype code specified in the application corresponds to the card Product's settings.

4 Authorization Scenarios

As has already been demonstrated, the contract account is a consolidation of subcontract accounts. The flow of funds through subcontract accounts is also reflected in the main contract accounts.

For bankcard contracts, an important parameter is Amount Available – the amount of funds that can be blocked during authorization. For a simple contract (not a subcontract) the value of this parameter is calculated as the following:

Own_Amount_Available = Current_Balance + Credit_Limit – Blocked_Amount

For subcontracts, their accounts are not considered to be independent and are considered as sub-accounts of the main contract. In this case, the system calculates Amount_Available from the subcontract's Own_Amount_Available value, which is calculated as the above, along with the Amount_Available value of the main contract.

The calculation method Amount_Available is determined by the *Auth Scenario* parameter, which can take on the following values:

- "Check" – see the section "Check, See Main".
- "See Main" – see the section "Check, See Main".
- "Billing Limit" – see the section "Billing Limit".

4.1 Check, See Main

Amount_Available calculation method when the *Auth Scenario* parameter is set to "Check":

Amount_Available = min(*Own_Amount_Available*, *Main_Amount_Available*),

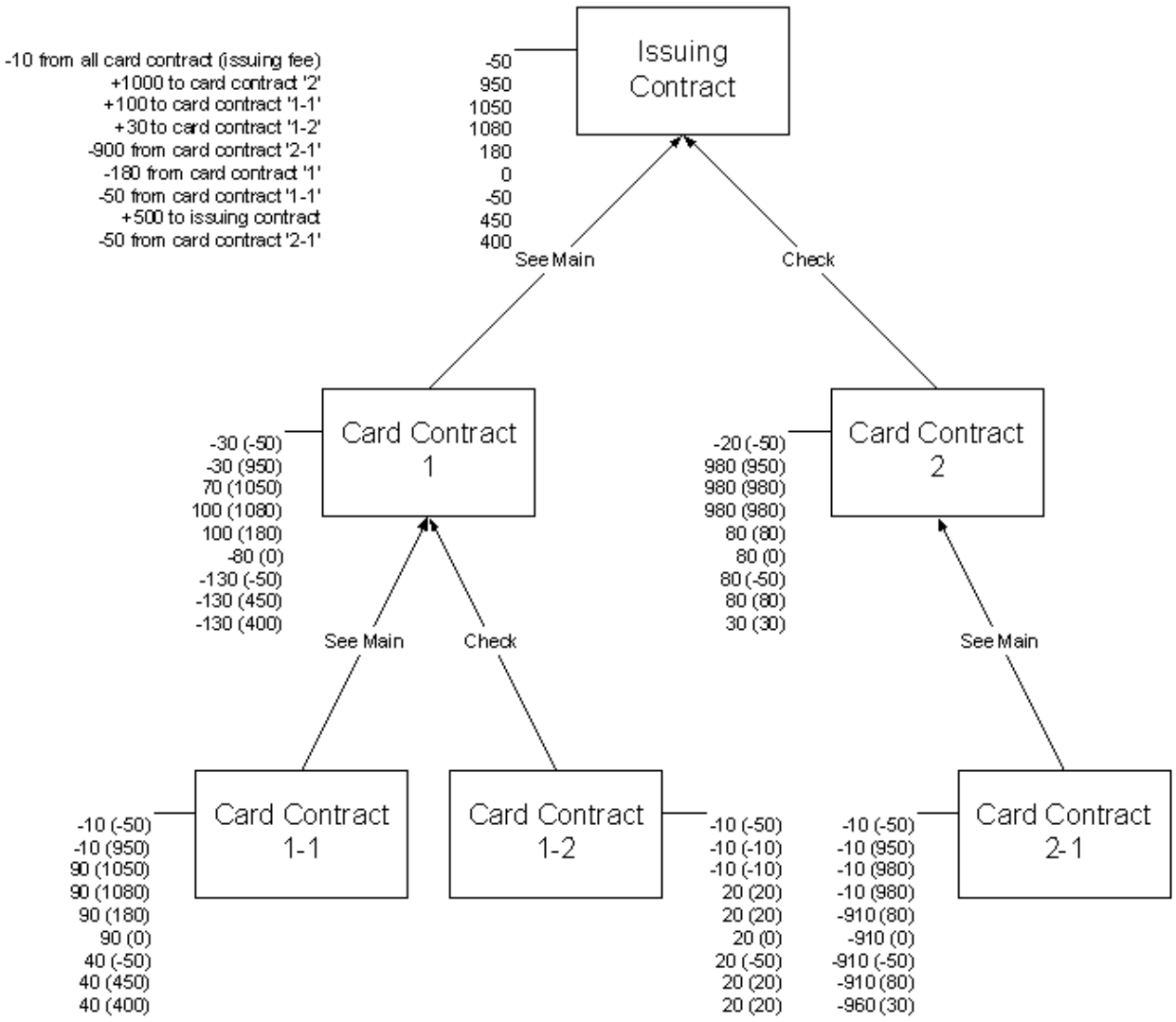
where Main_Amount_Available = Amount_Available of the main contract, calculated in the same way.

Amount_Available calculation method when the *Auth Scenario* parameter is set to "See Main":

Amount_Available = *Main_Amount_Available*,

where Main_Amount_Available = Amount_Available of the main contract, calculated in the same way.

To illustrate through a specific example:



Changes in a contract's amount available through various authorization scenarios

The upper left hand corner indicates contract actions and every contract displays next to it the state of the contract's "Client Deposit" account, and in parentheses, the amount available of the given contract after taking the appropriate actions.

These actions are described in detail here.

- For issuing a card, amount of 10 euro (EUR) was charged from every card contract. The account *Card Contract 1* is the consolidation of accounts *Card Contract 1-1* and *Card Contract 1-2*. So fund transfer from these subaccounts are reflected in the contract account *Card Contract 1* and furthermore, in the *Issuing Contract* account. The same occurs with the *Card Contract 2-1* and *Card Contract 2* accounts.
- Amount of 1000 EUR was transferred to the *Card Contract 2* account. This transfer was reflected to the consolidated contract account *Issuing Contract*. *Card Contract 2-1* is the subcontract of *Card Contract 2*, and therefore these funds are not reflected in the contract account. When two cards were issued for *Card Contract 2*, 20 EUR was charged and 980 EUR are remaining. As for the *Issuing Contract* account, only 950 EUR remain after 50 EUR was charged for issuing cards. So the *Issuing Contract*'s amount available is 950, the amount available of *Card Contract 2* is equal to the

minimum between 950 and 980, which is 950; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 950. The amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (950); the amount available of *Card Contract 1-1* is equal to the amount available of the main contract (950); the amount available of contract *Card Contract 1-2* is equal to the lesser of –10 and 950, that is, -10 EUR.

- The *Card Contract 1-1* account receives 100 EUR. This transfer is reflected in the consolidated contract accounts *Card Contract 1* and *Issuing Contract*. Now, the amount available of *Issuing Contract* is equal to 1050 EUR. The amount available of *Card Contract 2* is equal to the lesser of 980 and 1050 EUR, that is, 980; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 980 EUR; the amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (1050); the amount available of *Card Contract 1-1* is equal to the amount available of its main contract (1050); the amount available of the *Card Contract 1-2* is equal to the lesser of -10 and 1050, that is, -10 EUR.
- The *Card Contract 1-2* account receives 30 EUR. This transfer is reflected in the consolidated contract accounts *Card Contract 1* and *Issuing Contract*. Now, the amount available of *Issuing Contract* is equal to 1080 EUR. The amount available of *Card Contract 2* is equal to the lesser of 980 and 1080 EUR, that is, 980; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 980 EUR; the amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (1080); the amount available of *Card Contract 1-1* is equal to the amount available of its main contract (1080); the amount available of the *Card Contract 1-2* is equal to the lesser of 20 and 1080, that is, 20 EUR.
- 900 EUR were charged from the contract account *Card Contract 2-1*. This transfer is reflected in the consolidated contract accounts *Card Contract 2* and *Issuing Contract*. Now the amount available of the *Issuing Contract* is equal to 180 EUR. The amount available of *Card Contract 2* is equal to the lesser of 80 and 180 EUR, that is, 80; the amount available of *Card Contract 1-2* is equal to the amount available of *Card Contract 2*, that is, 80. The amount available of *Card Contract 1* is equal to the amount available of *Issuing Contract* (180); the amount available of *Card Contract 1-1* is equal to the amount available of the main contract (180); the amount available of *CardContract 1*-2 is equal to the lesser of 20 and 180, that is, 20 EUR.
- 180 EUR were charged to the contract account *Card Contract 1*. This transfer was reflected to the consolidated contract account *Issuing Contract*. Now, the amount available of the *Issuing Contract* is equal to 0. The amount available of *Card Contract 2* is equal to the lesser of 80 and 0 EUR, that is, 0; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 0 EUR; the amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (0); the amount available of *Card Contract 1-1* is equal to the amount available of its main contract (0); the amount available of the *Card Contract 1-2* is equal to the lesser of 20 and 0, that is, 0 EUR.
- 50 EUR were charged from the card contract account *Card Contract 1-1*. This transfer was reflected in the consolidated contract accounts *Issuing Contract* and *Card Contract 1*. Now, the amount available of the *Issuing Contract* is equal to -50. The amount available of *Card Contract 2* is equal to the lesser of 80 and -50 EUR, that is, -50; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, -50 EUR; the amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (-50); the amount available of

Card Contract 1-1 is equal to the amount available of its main contract (-50); the amount available of the *Card Contract 1-2* is equal to the lesser of 20 and -50, that is, -50 EUR.

- 500 EUR were charged to the *Issuing Contract* account. Now, the amount available of *Issuing Contract* is equal to 450 EUR. The amount available of *Card Contract 2* is equal to the lesser of 80 and 450 EUR, that is, 80; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 80 EUR; the amount available of *Card Contract 1* is equal to the amount available of the *Issuing Contract* (450); the amount available of *Card Contract 1-1* is equal to the amount available of its main contract (450); the amount available of the *Card Contract 1-2* is equal to the lesser of 20 and 450, that is, 20 EUR.
- 50 EUR were charged from the card contract account *Card Contract 2-1*. This transfer was reflected in the consolidated contract accounts *Issuing Contract* and *Card Contract 2*. Now, the amount available of the *Issuing Contract* is equal to 400. The amount available of *Card Contract 2* is equal to the lesser of 30 and 400 EUR, that is, 30; the amount available of *Card Contract 2-1* is equal to the amount available of *Card Contract 2*, that is, 30. The amount available of *Card Contract 1* is equal to the amount available of *Issuing Contract* (400); the amount available of *Card Contract 1-1* is equal to the amount available of the main contract (400); the amount available of contract *Card Contract 1-2* is equal to the lesser of 20 and 400, that is, 20 EUR.

When, "See Main" authorization scenario is used, at the closing of a billing cycle, the balance of the subcontract is zeroed out.

4.2 Billing Limit

In terms of calculating the Amount Available during an authorization, the "Billing Limit" scenario is the same as "Check". The only difference is the fact that the *Own Balance* of the contract is reset to zero at the end of the billing cycle.

For instance, credit limits are checked during authorization for the main contract and the subcontract, whereas payments are only made to the main contract. The system "restores" the individual credit limits of the subcontracts at the end of each billing cycle (minus the blocked amounts). Since the main contract's amount available is also checked during authorization, its subcontracts cannot exceed the shared credit limit. Therefore, no separate payments are needed to maintain the amount available in the main contract and its subcontracts.