

# Products and Contract Subtypes

# Contents

INTRODUCTION	1
CHAPTER 1. TERMS AND DEFINITIONS	2
CHAPTER 2. PRODUCTS	3
Entering Product Data	3
Main Product Parameters	3
Additional Product Parameters	5
Configuring Product Hierarchies	8
Copying Products	8
Copying Products within a Financial Institution	8
Copying Products to another FI	9
Copying Products to another Financial Institution with Inheritance of Properties	10
Inheritance of Product Properties	11
Event Setup	11
Entering and Editing Tags	13
Approving Products and Applying Product Changes to Contracts	14
Applying Product Changes to Contracts	15
Creating Product Groups	15
Configuring the "Contract Roles" List	16
Contract Classifiers	17
Setting a Default Classifier for a Product	19
"Configuration Groups" Classifiers	21
Configuring "Configuration Groups" Classifiers	21
Marking Products Using "Configuration Groups" Classifiers	23
Configuring Available Products	24
Changing a Product when Reissuing a Card	25
Redefining Contracts and Contract Accounts	26
Redefining a Fee Contract and Contract Account	26
Redefining an Interest Contract	31
CHAPTER 3. CONTRACT TYPES AND SUBTYPES	35
Contract Types	35
Entering Contract Subtype Data	37
Card Contract Subtypes Form	37
Account and Device Contract Subtype Forms	42
Checking Contract Subtype Fields	42
Configuring Preferred Counterparties	44
Configuring Message Templates (Group Msg)	44
Configuring the "Plastic Codes" Handbook	44
APPENDIX 1. AUTHORIZATION SCENARIOS	46
Check, See Main	46
Billing Limit	49
APPENDIX 2. TAGS IN PRODUCTS AND CONTRACT SUBTYPES	50



# Introduction

This document is intended for bank or processing centre employees responsible for configuring WAY4™ and describes Product creation and configuration.

While working with this document, it is recommended that users refer to the following reference material from WAY4 documentation series:

- Advanced Applications
- WAY4™ Accounting Schemes
- WAY4™ Service Packages
- Documents
- WAY4™ Global Parameters
- Issuing Module User Manual
- Acquiring Module User Manual
- WAY4™ Advanced Tariff Management
- Preferred Contractors
- WAY4™ Client and Contract Classifiers

The following conventions are used throughout this document:

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

## Chapter 1. Terms and Definitions

WAY4 is used to:

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

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

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

Contract properties are determined by three main parameters:

- Contract type/subtype – determines the "nature" of a contract: card contract (MasterCard or Visa, magnetic stripe or smart card), device contract (ATM, POS terminal, imprinter), or issuing/acquiring contract (a set of contract accounts and rules for working with them for a natural person, a legal person, or a bank branch)
- Service Package – contains a list of transactions, fee parameters, transaction processing rules, and usage limiters
- Accounting Scheme – determines a contract's accounts, their properties and relations between them (see Fig. 1)

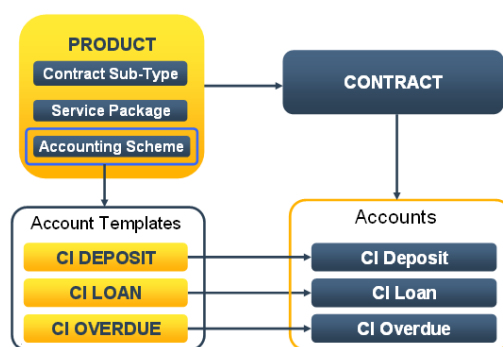


Fig. 1. Relations between system objects

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

## Chapter 2. Products

A Product is used as a template to register new contracts or to change properties of contracts already registered in WAY4 using Products.

Using 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 Advanced Applications Module Administrator Manual) when applications to issue cards are entered.

## Entering Product Data

### 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 Fig. 2).

Institution	Client	Product	Contract	Product Group	Name	Acc Scheme	Tariff Domain	Is Ready	Product Template	Contr Subtype	Service Pack	Report Type	Date Scheme	Code
Principal	Private	Issuing	Account	Issuing Debit	Issuing	001-Debit Iss USD Priv		Ready		001-Private Client	001-Accounting Private			5678
Principal	Private	Issuing	Card	Issuing Debit	Issuing	001-Full Iss Local Priv		Not Ready		001-VISA CLASS	001-Private Client			SMOKE

Fig. 2. Form for entering product data

Products are grouped by a number of criteria:

- Field *Institution* – financial institution registered in WAY4 for which the Product is created
- Field *Client* – client type:
  - "Private" – natural person
  - "Commercial" – legal person
  - "Accountant" – bank branch
- Field *Product* – drop-down list of Product categories ("Issuing", "Acquiring", "Bank Accounting", or "Accounting" (the last value is only used for backward compatibility)).
- Field *Contract* – Contract category for which the Product may be used (Card, Account, Device)
- Field *Product Group* – name of a Product group registered in the system (see "Creating Product Groups"); used in the Advanced Applications module
- Field *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. If the bank's distribution package does not include

the WAY4 Tariffs module, the field is empty. For more details, see the WAY4™ Advanced Tariff Management User Manual.



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

- *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.
- Field *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* – Accounting Scheme that determines a set of accounts and the way they interact (see the WAY4™ Accounting Schemes Administrator Manual).
- *Contract Subtype* – contract subtype (see "Contract Types and Subtypes")
- *Service Package* – Service Package that determines a set of authorised operations, Product fees, etc. (see the WAY4™ Service Packages Administrator Manual).
- *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 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 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 and 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 "Configuring Product Hierarchies").

The [Copy To] button is used to copy a Product to another financial institution (see "Copying Products to another FI").

The [Full Info] button is used to access the form with additional Product information (see "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 "Contract Classifiers").

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".

## Additional Product Parameters

To access additional Product parameters, click the [Full Info] button in the "Products" form (see Fig. 2 in section "Main Product Parameters"). This will open the "Full Info for <name of Product>" form (see Fig. 3) used to specify additional Product parameters.


Fig. 3. 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 details, see "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 "Issuing Module" document, the section "Creating New ATM Retail Contracts" of the "Acquiring Module" document. 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 details, see "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 "Tags in Products and Contract Subtypes".



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 authorisation (for more details, see "Authorization Scenarios")
- *Usage Scenario* – the value of this field determines how usage limiters are applied during authorisation request processing (for more details, see the Usage Limiters Administrator Manual):
  - "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 delivery package includes the Advanced Tariff Management module. For more information, see the document "WAY4™ Advanced Tariff Management".



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

The [Duplicate] button is used to copy a Product (see "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 Accounting Scheme.

The [Tagged Data] button is used to optimise 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 "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 "Contract Classifiers".


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 "Contract Classifiers").


## 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 Fig. 2 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 Accounting Scheme as the parent Product. The Accounting Scheme is automatically copied to the child Product's properties. When users attempt to change the Accounting Scheme, a warning message is generated.

For more details on contract relation types, see section "Contract Hierarchy" in the Issuing Module User Manual.


## 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.

### 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 Fig. 3 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 Advanced Tariff Management module is not included in the WAY4 basic configuration and is supplied according to an additional agreement with the WAY4 vendor.
- After copying, a new Product is added in the list of Products in the "Products" form (see Fig. 2 in section "Main Product Parameters").

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

## Copying Products to another FI

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

- In the "Products" form (see Fig. 2 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 Fig. 4), select the FI to which the Product will be copied and click the [Proceed] button.

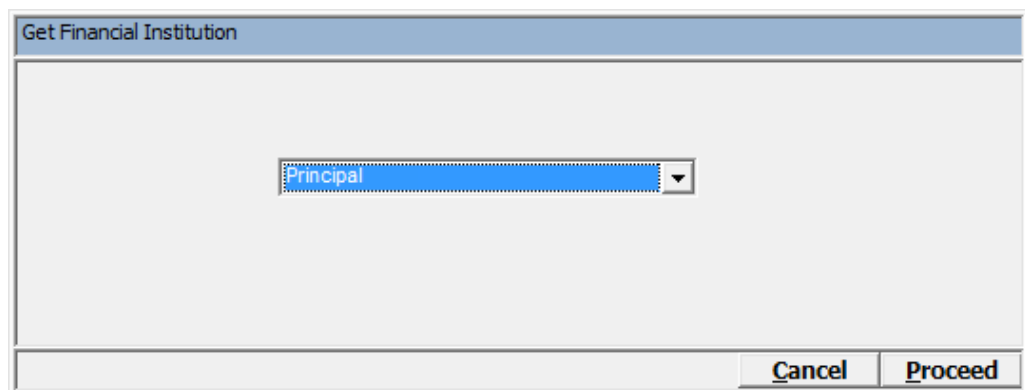


Fig. 4. Form for selecting an FI

If when copying a Product it turns out that there is already a Product with this code in the target financial institution, existing Product components (subordinate

Products) remain unchanged. New subordinate Products are added (subordinate Products in the Product being copied and absent from the Product in the target financial institution).

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".

## Copying Products to another Financial Institution with Inheritance of Properties

To make it easier to synchronise parameter changes Products belonging to different financial institutions, it is possible to make changes in one template Product with subsequent automatic changes to parameters in other Products related to the template Product (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 Fig. 2 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 Fig. 4 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 is created (Product hierarchy) that inherits the properties of the original Product (Accounting 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.

## Inheritance of Product Properties

Product characteristics may be inherited by sub-products from main products. For this, the *Set Recur* 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.

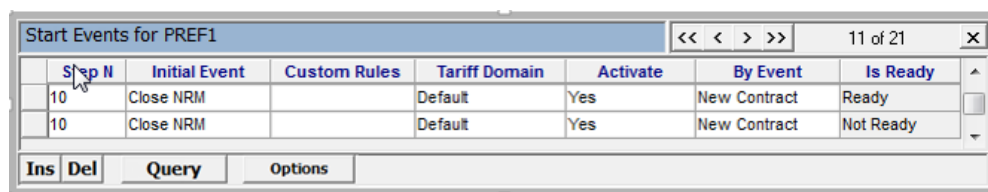
 When the *Recur From* field is used, all Product properties are inherited (except Product status and *Date From/Date To* dates). It is not recommended that this field be used without consulting with the WAY4 system vendor's representatives.

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

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

## Event Setup

Events for contracts using this Product are set up in the "Start Events for <name of Product>" form (see Fig. 5). To access the form, click the [Start Events] button in the additional Product data form.




Step N	Initial Event	Custom Rules	Tariff Domain	Activate	By Event	Is Ready
10	Close NRM		Default	Yes	New Contract	Ready
10	Close NRM		Default	Yes	New Contract	Not Ready

Fig. 5. 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 Events Administrator Manual).
- *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 Issuing Module User Manual). 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 Events Administrator Manual). Event opening only changes a tariff domain when a domain code is specified on the Product level.



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

- "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 "Contract Classifiers").



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 Parm*s field of the corresponding Event type.
- ◆ An Event type with the `PRODUCT_OPTION;` tag cannot be used in the "Start Event" 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 "Contract Classifiers").

## Entering and Editing Tags

The "Full Info for <name of Product>" form (see Fig. 3 in section "Additional Product Parameters") contains special field for entering and editing tags – *Custom Data*.

The "Tagged Data" form (see Fig. 6) is used to optimise 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.

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

Fig. 6. Form "Tagged Data"

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

- A tag name in the *Tag* field can be selected from the system list of tags. If a tag is absent from the list, its name can be entered from the keyboard.
- In the *Value Data* field, enter a tag value.
- Field *Value Tag*:
  - When entering tag parameters in the "Tagged Data" form, select the "Tag Present" value in the *Value Tag* field so that tag parameters are saved correctly. After data is saved, tag parameters will be displayed in the corresponding field of the parent form (in field *Special Parms* or *Template Details*).
  - When the "Tag Absent" value is selected, the tag record will be deleted from the "Tagged Data" form as well as from the corresponding field of the parent form after changes are saved.
- The *Value Type* field is used to determine the type of the tag value entered in the *Value Data* field:
  - "CheckBox" – in this case, it is not necessary to fill in the *Value Data* field
  - "String" – string tag value
  - "Counter" – the value of the tag must be an integer (from "0" to "9")
  - "Tag" – the value of the tag must be either "Y" or "N"
  - "Money" – numeric tag value
  - "Currency" – the value of the tag must be a numeric currency code
  - "Unknown" – no tag value type is specified, but the *Value Data* field must be filled in



- "List" – the tag value can be set as a list of values. In initial manual setup, a comma-delimited list of values is specified in the *Value Data* field (or in the *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 show results of tag parameter check:
  - The field contains the "Ready" value if the check is successful.
  - The field contains the "Not Ready" value if errors have been detected during the check.

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

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

## 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 and 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.

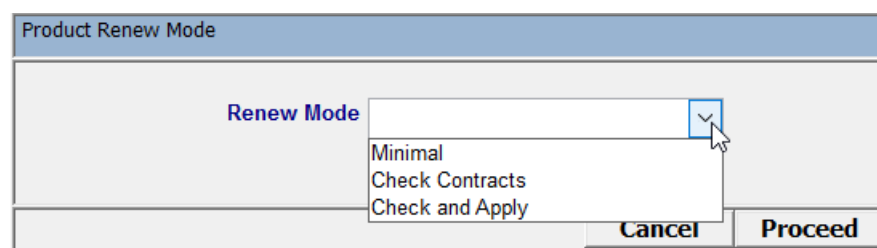


Fig. 7. "Product Renew Mode" form



If errors are found during checks, the corresponding 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".

Products can be approved in the form "Products (No Hierarchy)" (Full → Configuration Setup → Products → Product Definition → Products (No Hierarchy)). This form shows all Products registered in WAY4, including subordinate Products in a "Main/Sub" hierarchy. I.e. all Products requiring approval, regardless of hierarchy, can be filtered in the form.

Approval is performed as with the "Products" form. If the approval procedure is called for a subordinate Product:

- 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 *Is Ready* field contains "No"), approval is called for the parent Product, and respectively, for the entire hierarchy (for all branches under the parent Product).

## 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 details, see the "APPROVE\_IMMEDIATE" section in the WAY4™ Global Parameters Administrator Manual.



Starting from version 38.30, when approving contracts after changing an Accounting Scheme or Product, the correspondence of existing accounts to those of the new Accounting 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 Accounting Scheme was changed).

## Creating Product Groups

Users may group issuer products in WAY4. Product groups are used during application import using the Advanced Applications module (see the Advanced Applications Module Administrator Manual).

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 Fig. 8).

Product Category	Name	Code
Issuing	Issuing Debit	ISSDEB
Acquiring	Acquiring Retail	ACQRET

Fig. 8. Grid for entering and editing Product group data

The grid contains the following fields:

- *Product Category* – select from a list of product categories:
  - "Issuing" – Products for issuing contracts
  - "Acquiring" – Products for acquiring contracts
  - "Accounting" – Products for banking system contracts. This field is used for backward compatibility.
  - "Bank Accounting" – Products for bank contracts
- *Name* – Product group name
- *Code* – Code used when downloading external files into the system to link downloaded 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 "Main Product Parameters").

## 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 Fig. 9.

Name	Code	Product Category	Contract Category	Main Card Flag	Group Code
Main Card	345	Issuing	Card	Yes	CONTRACT_ROLE

Fig. 9. 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.

## Contract Classifiers

Configuration of contract classifiers in a Product allows set classifier values to be automatically assigned to contracts when they are created (contracts created on the basis of the corresponding Product).


The list of contract classifiers in a Product is configured in the "Classifiers for <наименование Продукта>" form (Full → Configuration Setup → Products → Product Definition → Products → [Classifiers]), see Fig. 10.

Classifier	Default Value	Editable	Set To Contract	Active Date From	Active Date To	Is Ready
Product Based 1	value_1	No	No	00/00/0000	00/00/0000	Ready


Fig. 10. Configuration of contract classifiers in a Product

This form contains the following fields:


- *Classifier* – name of the contract's user classifier.

 The entire list of classifiers (client and contract system classifiers and user classifiers) is shown in this field's list. Only contract user classifiers should be selected from this list. When a client or contract system classifier is selected or a client user classifier, this setting will not function.

- *Default Value* – the classifier value that is assigned to a contract/client when a contract is created.

 The *Default Value* field is mandatory. If the *Default Value* field is left empty, the classifier set in the *Classifier* field will not be assigned to the contract.

- *Editable* – this field is reserved for future use.
- *Active Date From* – period start date (the period end date is specified in the *Active Date To* field). Contracts created during this period for a given Product will be assigned the classifier set in the *Classifier* field with the value from the *Default* field.
- *Active Date To* – period end date (the period start date is specified in the *Active Date From* field). Contracts created during this period for a given Product will be assigned the classifier set in the *Classifier* field with the value from the *Default* field.

 When checking the active period of a classifier set in a Product, the *Active Date From* and *Active Date To* dates are compared with the financial institution's banking date. The active period for the classifier value is determined by the system date (*Date From* and *Date To* fields in the "Values for <classifier name>" form; "Full → Configuration Setup → Common Handbooks → User Classifiers → [Values]").

- *Set To Contract* – this field defines rules for working with the default classifier value set in the *Default Value* field:
  - "Y" – if the field is not filled in or "Y" is specified, the value of the *Default Value* field is set in the contract when approving the contract. This is the default behaviour.
  - "N" – the value of the *Default Value* field is NOT set in the contract when approving the contract. This value will be considered as the default value for this Product and can be used as follows:
    - ◆ The default value of the classifier in the Product can be used if a classifier is not set in the contract.
    - ◆ For the default value set for the Product in the "Classifiers for" form to be used when there is no classifier in the contract, set the DEFAULT\_FROM=PRODUCT; tag in the classifier's *Add Info* field.
    - ◆ If a contract parameter with the "Product" value of the *Value Location* field is used, the "Classifier" value can be specified in the *Check Product* field of this contract parameter. In this case, the value of the classifier is determined according to the PRODUCT\_OPTION table, not the APPL\_PRODUCT table (that is, the default value set for the Product in the "Classifiers for" form is used).

For more information, see the section "Contract and Client Custom Parameters" of the document "WAY4™ Client and Contract Classifiers".



Note that a classifier with the "N" value in the *Set To Contract* field is added in the "Products" form (Full → Configuration Setup → Products → Product Definition → Products) using the "Set Default Classifier Value" command of the [Approve] button's context menu. The [Ins] button should not be used to add this classifier.

- *Is Ready* – this field shows whether changes made in this record were approved (when the Product is approved).

Clicking on the [Classifiers] button in the "Classifiers for <Product name>" form (see Fig. 10) opens a form with this classifier's parameters. Click on the [Value] button to view the list of values for this classifier. For classifiers with the "N" value in the *Set To Contract* field, the list of Products ([Prod defaults] button) and financial institutions ([FI defaults] button) for which this is the default classifier can be viewed in this form (see the section "Setting a Default Classifier for a Product" and "Setting a Default Classifier for a Financial Institution" of the document "Financial Institutions").

"Classic" user classifiers for contract segmentation and user classifiers to include Product options can be used in a Product. Product options are indicators of certain events or stages in a contract's lifecycle. For example, allowing/prohibiting authorisation for a contract, etc. A product option may be "blocked". For example, authorisation for a contract may be blocked when the contract has lengthy delinquency. For more information, see the section "Blocking Product Options" of the document "WAY4™ Client and Contract Classifiers".

## Setting a Default Classifier for a Product

A default classifier and classifier value for a Product are set/changed in one of the following ways:

- For a specific Product – using the "Set Default Classifier Value" command from the context menu of the [Approve] button in the "Products" form (Full → Configuration Setup → Products → Product Definition → Products). This command opens the "Set Classifier Value" form, see Fig. 11.

Fig. 11. Configuring a default value for a Product

In the "Set Classifier Value" form, fill in the following fields:

- *Classifier* – from the list, select a user classifier for the contract in which the DEFAULT\_FROM=PRODUCT; tag is set.
- *Value* – classifier value that will be used as the default value for the Product.
- *Add Info* – reserved for forward compatibility.
- *Date From* and *Date To*- start/end date of the period during which this default value is active.

When the [Proceed] button in the "Classifiers for <Product name>" form is clicked on (see Fig. 10 in the section "Contract Classifiers"), a new record will be created with the *Default Value* field filled in and the "N" value in the *Set To Contract* field.

If a default classifier had been set earlier for a Product, the *Active Date To* field will automatically be filled in for the "old" classifier record (according to the period specified for the new default classifier). When this date arrives, the record will be closed and the "Closed" status will be set in the *Is Ready* field.


- For Product groups – to do so, filter Products in the "Products (No Hierarchy)" form (Full → Configuration Setup → Products → Product Definition → Products (No Hierarchy). To assign a default classifier to all Products shown in the form, execute the "Set Default Classifier Value (For All)" command. After clicking on the [OK] button in warning message "Do you want to execute 'Set Default Classifier Value (For All)' action for all records?", a form for entering default classifier parameters will open. This form is similar to that shown in Fig. 11. Fields are filled in like when adding a default classifier for a specific Product.

When the "Set Default Classifier Value"/"Set Default Classifier Value (For All)" command is used to add or change a default classifier/value, Product

status remains unchanged (Is Ready), i.e. the Product does not have to be approved after this action.

Adding/changing the default classifier for a Product does not cause any changes for specific contracts.

For contracts for which a classifier is not set and if the DEFAULT\_FROM=PRODUCT tag is set for a classifier, the classifier value is taken from Product parameters, and if it is absent, the default value from the parameters of the classifier itself is used (value of the *Default Value* field in the "User Classifiers" form).

 A default classifier and classifier value can be set for a financial institution. See the section "Configuring a Default Classifier for a Financial Institution" of the document "Financial Institutions". The value of a default classifier for a Product has a higher priority than that for a financial institution.

## "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.

## Configuring "Configuration Groups" Classifiers

To set up "Configuration Groups" classifiers, select "Full → Configuration etup → 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 Fig. 12).



Name	Code	Scope
Credit	CREDIT	STAT

Fig. 12. Form "Configuration Groups"



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

- *Name* – group name
- *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 Fig. 13) used to set up classifiers.

Classifier Name	Classifier Code	Table Code	Add Info	Scope
Card Brands	CARD_BRAND	APPL_PRODUCT		STAT

Fig. 13. 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 Fig. 14) used to set up classifier values.

Classifier	Value	Value Code	Parent Value
Card Brands	MasterCard Int.	ME	
Card Brands	MC Platinum	MPL	Card Brands: MasterCard Int.
Card Brands	MC Gold	MGOLD	Card Brands: MasterCard Int.
Card Brands	MC Maestro	MAE	Card Brands: MasterCard Int.

Fig. 14. 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 Fig. 15), click the [Downwards] button.

Classifier	Value	Value Code	Parent Value
Card Brands	MC Platinum	MPL	Card Brands: MasterCard Int.
Card Brands	MC Gold	MGOLD	Card Brands: MasterCard Int.
Card Brands	MC Maestro	MAE	Card Brands: MasterCard Int.

Fig. 15. 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.

## 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 Fig. 16) 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 Fig. 13 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.

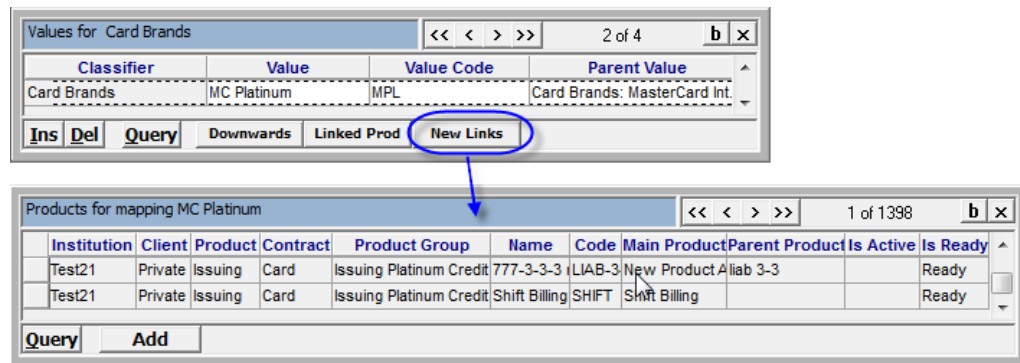


Fig. 16. Marking data with "Configuration Groups" classifiers

- Clicking the [Linked Prod] button (see Fig. 16) 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 in the additional Product data form form:
  - To mark data in the additional Product data form (Full → Configuration Setup → Products → Product Definition → Products → [Full Info]) click the [Mapping] button.
  - In the "Mapping for <Product name>" form that opens (see Fig. 17) 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.

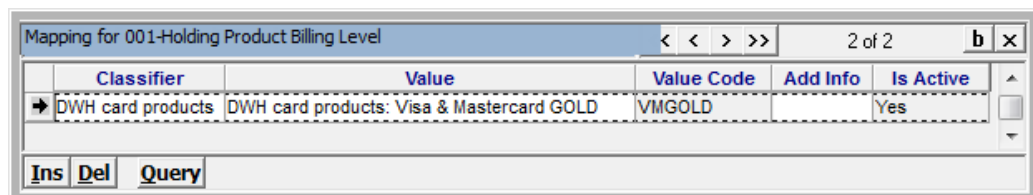


Fig. 17. "Classifiers for <Product name> form"

## 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 Fig. 28.

Status	Value	Custom Rules
Client Type	Private Resident	
Client Role	Main	

Ins Del Query

Fig. 18. "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".

## 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 codet> 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.

AND

- ♦ An individual limiter with this code was active for the old contract before reissue.

## Redefining Contracts and Contract Accounts

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

### Redefining a Fee Contract and Contract Account

#### Overview

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.

A fee contract and contract account can be redefined for the main Service or for a Custom Fee:

- For the main Service, the contract and account to which a fee is transferred can be redefined (i.e. the *Fee Contract* and *Fee Account* fields in the Service). The FEE\_CONTRACT and REDEF\_FEE\_ACC\_CODE tags set in the Service are used to do so.
- For a Custom Fee, the following can be redefined:
  - The contract and account to which the fee is transferred (i.e. the *Fee Contract* and *Fee Account* fields in the Service). The FEE\_CONTRACT and REDEF\_FEE\_ACC\_CODE tags set in the Service are used to do so.
  - The contract and account from which the fee is charged. The CONTRACT\_FOR and REDEF\_ACC\_CODE tags set in the Service are used to do so.

The FROM\_DECISION:<decision code>; and FROM\_TARIFF:<tariff type code> values of the FEE\_CONTRACT, REDEF\_FEE\_ACC\_CODE, CONTRACT\_FOR, and REDEF\_ACC\_CODE tags are used in redefinition. I.e. decisions or tariffs can be used for redefinition. 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.

### Usage Guidelines

Usage guidelines for 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 a FROM\_TARIFF value for the following:
  - If selection of a fee contract depends on Product settings – on the Product that is used, tariff plan, Service Package, etc.
  - To optimise 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).

### 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 Fig. 19 – when a transaction is made using this Service, the fee contract (001-MERCHANT\_FEE) can be redefined using a decision with the code "doc3".

The screenshot shows a 'Posting' form with the following fields and values:

- Fee Code: ddd
- Floor Limit: 0,00
- Value Days: 0
- Service Allowed: Always
- Service Details: FEE\_CONTRACT=FROM\_DECISION:doc3:STRICT;
- Account Type: (empty)
- Account Curr: (empty)
- Contract Type: Our POS
- Fee Contract: 001-MERCHANT\_FEE
- Fee Account: Retail Fees Passive
- Ready button

A red circle highlights the 'Service Details' field, and a blue arrow points from the 'Fee Contract' field to it.

Fig. 19. 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. I.e. the prefix of the corresponding financial institution (institution to which the contract with which the transaction being made (Target or Source contract) belongs is automatically attached to it. 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 Fig. 20

Fig. 20. 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).

### 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 Fig. 21, the bank contract from the *Fee Contract* field (001-MERCHANT\_FEE) can be redefined using a tariff with the TRF\_4\_FEE\_CONTRACT code.

The screenshot shows a 'Posting' form with the following fields and values:

- Fee Code: ddd\_trf
- Floor Limit: 0,00
- Value Days: 0
- Service Allowed: Always
- Service Details: FEE\_CONTRACT=FROM\_TARIFF:TRF\_4\_FEE\_CONTRACT;
- Account Type: (empty)
- Account Curn: (empty)
- Contract Type: Our POS
- Fee Contract: 001-MERCHANT\_FEE
- Fee Account: Retail Fees Passive
- Ready button

A red box highlights the 'Service Details' field, and a blue arrow points from the 'Fee Contract' field to it.

Fig. 21. 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. I.e. the prefix of the corresponding financial institution (institution to which the contract with which the transaction being made (Target or Source contract) belongs is automatically attached to it. 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;

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

Tags used to check document parameters (fDOC\_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 Fig. 23.



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 Fig. 22).



Tariff Domain	Tariff Role	Tariff Type	Name	If Limit Tariff	If Preference Type	If Currency	If Event Type	Apply Rules
SUB TARIFF DOMAIN	Redefinition	TRF_4_FEE_CONTRACT	TARIFF_FOR_FEE_CONTRACT					

Ins Del Query Approve Redefinition

Redefinition for TARIFF\_FOR\_FEE\_CONTRACT

Start Date	Code	Apply Mode	Is Ready
02/01/2007	999-VISA_FEES	From Tariff	Ready

Ins Del Query

Fig. 22. Configuring a tariff to redefine a fee contract, without analysing document tags

Tariff Domain	Tariff Role	Tariff Type	Name	Unit	Terenceurrency	Apply Rules
MAIN TARIFF DOMAIN	Redefinition	TRF_4_FEE_CONTRACT	FEE_CONTRACT FOR CARD			DOC_TAG=SOURCECH;SOURCE_CHANNEL=P.

Ins Del Query Approve Redefinition

Redefinition for FEE\_CONTRACT FOR CARD redefinition

Start Date	Code	Apply Mode	Is Ready
02/01/2007	999-CLIENT_FEE	From Tariff	Ready

Ins Del Query

Fig. 23. 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").

## Redefining an Interest Contract

### Overview

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 Accounting 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.

### 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 optimise Product settings – the FROM\_TARIFF construction makes it possible to reduce the number of account templates that are configured.

## Using Decisions for Redefinition

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

- In the Accounting Scheme or account template, set the INTEREST\_CONTRACT=FROM\_DECISION:<decision code>; tag (see Fig. 24). 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.

Interest Contract	001-INTERESTS
Interest Accrual Account	Deposit Int Accrual
Interest Exp/Rev Account	
Supplementary Credit Acc	
Supplementary Debit Acc	
Interest Tariff	

GL Properties	
GL Type	
GL #	E/INTST-DEPOSIT
Numeration Type	First Transfer
Aggregate GL For	
Off-balance XF Account	
FX Type	
GL Tariff	

Reference Mode	
Template Details	INTEREST_CONTRACT=FROM_DECISION:t_interest:STRICT;

Fig. 24. Configuring redefinition of an interest contract using a decision

- **i** The value of the INTEREST\_CONTRACT tag in an account template has a higher priority.

The INTEREST\_CONTRACT tag value set in an Accounting Scheme works for all the scheme's templates. If a particular template has its own INTEREST\_CONTRACT tag, this 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 Fig. 25.

- **i** By default, the contract specified in this field is used as a template. I.e. the prefix of the corresponding financial institution (institution to which the contract with which the transaction being made (Target or Source contract) belongs is automatically attached to it. 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 Fig. 25 depending on the value of the INTEREST\_CONTRACT contract classifier).

The screenshot displays the configuration of a decision for redefinition of an interest contract. It consists of three main panels:

- Decision Panel:** Shows a table with columns: Name, Code, Group Code, Default Result, Applies To, Client Cat, Contract Cat, Product Cat, Add Info, and On Off Mode. The entry is 'test decision for interest contract' with code 'test\_interest' and 'Any Contract'.
- Rules for test decision for interest contract Panel:** Shows a table with columns: Priority, Rule Category, Condition Type, Result, Comment Text, Classifier, Apply Rules, Is Active, Date From, and Date To. The entry is '0' with 'By Condition' category, 'AND' condition type, and result '001-INTERESTS PRD'.
- Conditions for Rules for test decision for interest Panel:** Shows a table with columns: Classifier, Condition Rule, Classifier Value, and Apply Rules. The entry is 'INTEREST\_CONTRACT' with condition rule 'IS' and classifier value '999-INTEREST'.

Arrows indicate the flow from the 'Rules' table to the 'Conditions' table and then to the 'Results' table.

Fig. 25. 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".

### 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 Accounting 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 Fig. 26 – with the "t\_interest" tariff type code).

The screenshot displays the configuration of redefinition of an interest contract using a tariff. It includes a 'GL Properties' section with the following fields:

- Interest Contract: 001-INTERESTS
- Interest Accrual Account: Deposit Int Accrual
- Interest Exp/Rev Account:
- Supplementary Credit Acc:
- Supplementary Debit Acc:
- Interest Tariff:
- GL Type:
- GL #: EV/NTTST-DEPOSIT
- Numeration Type: First Transfer
- Aggregate GL For:
- Off-balance XF Account:
- FX Type:
- GL Tariff:

A red circle highlights the 'Template Details' field, which contains the text: INTEREST\_CONTRACT=FROM\_TARIFF:t\_interest:STRICT;

Fig. 26. Configuring redefinition of an interest contract using a tariff decision

**i** The INTEREST\_CONTRACT tag value in an account template has higher priority.

The INTEREST\_CONTRACT tag value set in an Accounting 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.

**i** By default, the bank contract number specified in this field is used as a template. I.e. the prefix of the corresponding financial institution (institution to which the contract with which the transaction being made (Target or Source contract) belongs is automatically attached to it. 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. Например, INTEREST\_CONTRACT=FROM\_TARIFF:t\_interest:STRICT; (see Fig. 26).

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

**i** It is not mandatory to configure conditions for selecting a tariff (checking document parameters, etc.). I.e. a contract can be redefined simply when the it uses a specific tariff plan, without additional conditions (see Fig. 27).

The screenshot shows a software interface for configuring tariffs. The main window is titled 'Tariff Data Input By Tariff'. It contains a table with columns: Tariff Domain, Tariff Role, Tariff Type, Name, Limit, Tariff Reference, Current, Event Type, Apply Rules, and Is Ready. A row is selected with 'Main Tariff Domain', 'Tariff Redefinition', 't\_interest', 'Redefine Interest Contract', and 'Ready'. Below this table are buttons: 'Ins', 'Del', 'Query', 'Approve', and 'Redefinition'. The 'Redefinition' button is active. A sub-window titled 'Redefinition for Redefine Interest Contract' is open, showing a table with columns: Start Date, Code, Apply Mode, and Is Ready. A row is selected with '24/03/2016', '999-INTEREST', 'From Tariff', and 'Ready'. The 'Code' field is highlighted with a red circle. Below this table are buttons: 'Ins', 'Del', 'Query', and 'Cancel'.

Fig. 27. Configuring redefinition of an interest contract using a tariff

## Chapter 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.

### Contract Types

The WAY4 System 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 Fig. 28 shows the "Card Contract Types" grid used for creating card contract types.

Card Contract Types				
		<< < > >>		1 of 18
	Name	Channel	Service Code	RBS Code
→	Cirrus/Maestro	MasterCard (IPM)		CM
	EuroCard/MasterCard	MasterCard (IPM)		EC
	Electron/Plus	VISA		PC
	Our EuroCard/MasterCard	Our EC Cards	101	ec
	Our Cirrus/Maestro	Our EC Cards	121	cm
	Our Electronic Purse	Internal		EP
	Our Local Card	Internal	501	lc
	Our VISA Cards_1	Our VISA Cards	101	vc1
	VISA Cards	VISA		VC
	AMEX Cards	AMEX		AC
	DCI Cards	Diners Club		DC
	JCB Cards	JCB		JC
	Our Electron/Plus	Our VISA Cards	121	pc
	Our JCB Cards	Our JCB Cards		jc
	SIBS	VISA Unibanco Mod B C		si
	Our VISA Cards	Our VISA Cards	101	vc
	Our AMEX Cards	Our AMEX Cards		ac
	Our CUP Cards	Our CUP Cards		hc

Ins Del Query Actions... SubTypes Msg Dict

Fig. 28. 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* – Service code of 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 command "Validate" of the [Actions] button is used to check the *RBS Code* field (a check is made that the field is filled in and the value of the field is checked for uniqueness), 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").

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 "Entering Contract Subtype Data"). After copying, it is necessary to set up the following objects for the new contract type:

- Transaction subtypes (see section "Transaction Subtypes" in the Documents Administrator Manual).
- New Services in the existing Service Package (if necessary)
- New Service Packages for the new contract type (see the WAY4™ Service Packages Administrator Manual).

The [SubTypes] button is used to create and configure contract subtypes (see "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 details, see section "Message Dictionary" in the WAY4™ Dictionaries Administrator Manual).

## 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 Fig. 31 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 [SubType] in the "<Contract Type> Contract Types" form.
- Click [Ins] to add a blank record in the "SubTypes for <Contract type>" form (see Fig. 29 in section "Card Contract Subtypes Form") and fill in all the record fields one by one.

### Card Contract Subtypes Form

The Fig. 29 shows the form "SubTypes for <name of card contract type>".

Institution	Client	Name	Is Active	BIN	Min #	Max #	Channel	BIN Record	IPS Product
Principal	Private	001-VISA GOLD	Yes	401550	0100000000	0199999999	Our VISA Cards	73091	
Principal	Private	001-VISA CLASS	Yes	401550	0100000000	0199999999	Our VISA Cards	73091	

Fig. 29. 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 set up preferred contractors (see "Configuring Preferred Counterparties").
- The [Mapping] button is used to mark contract subtypes using "Configuration Groups" classifiers. For more details, see section "Contract Classifiers".
- 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", and "Checking Contract Subtype Fields").

The card contract subtype grid (see Fig. 29) contains the following fields:



**Institution**

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

**Client**

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

**Name**

The *Name* field is the contract subtype name; the following name format is recommended: "NNN–name", 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 "Financial Institutions" form; to display the form, choose "Full → Configuration Setup → Main Tables → Financial Institutions" on the user menu). This field's value must be unique within the contract type.

**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 system forbids operators to create new contracts for that subtype.

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.

**BIN**

The *BIN* field is used to create BINs affiliated with a payment system. This field value is used during automatic contract numbering as the first several digits of the number.



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 centre.

**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.

**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.

**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 executed. A search is made for a record in the BIN table corresponding to the sub-type. 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. I.e. a search is first made by six characters (maximum match), then by four characters, etc. If the search by three characters does not produce a result, it is considered that the BIN table record was not found <https://ows-jira/browse/CARDS-50670>.


 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 *BIN Record* field of the contract subtype, 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 sub-type's *BIN Record* field is automatically filled in:

- When approving a Product with a sub-type 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").

### **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 executed (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").

### **Expire For New**

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

### **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.

### **Service Code**

The *Service Code* field contains the service code of the contract subtype cards 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 Fig. 29 in section "Card Contract Subtypes Form"). The form's list of values is generated using the handbook "Card Service Codes" Full → Configuration Setup → Contract Types → Card Service Codes".

### **PM (Pin Management) 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. To open the form, choose "Full →

Configuration Setup → Card Production Setup → Bank Production Parameters" on the User Menu, select the required bank record and press [Parameters]. For more information about card production parameters, see the section "Changing the Card Production Parameters within one PAN Range" of the document "Configuring WAY4™ for Magnetic Stripe Card Issuing".

### **Numeration Type**

Allows numeration rules to be set for a card contract:

- "Custom" – the individual part of the contract number is generated using a custom procedure.
- "Random" – the 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" – the individual part of the number is generated sequentially in ascending order by adding a unit to the number specified in the *Current #* field.
- "Manual" or any other value differing from those listed above – manual entry of a number.

### **Validation Type**

This field is used to specify special parameters in the form of tags. For more details, see "Tags in Products and Contract Subtypes".

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

### **Plastic Code**

This is the plastic design code used to produce plastics for the contract subtype. 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".

### **RBS Code**

This 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.

### **Add Params**

This field is used to specify special parameters of contract subtypes in the form of tags. For more details, see "Tags in Products and Contract Subtypes".

### **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.

## Account and Device Contract Subtype Forms

The grids for account and device contract subtypes are similar to the card contract subtype grid (see Fig. 29 in section "Card Contract Subtypes Form") with the following differences:

First, these forms do not contain fields typical of card contract subtypes (e.g. *BIN* and *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", "Current #".

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

### **Prefix**

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

### **Num Type**

This field indicates the contract numeration type:

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



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 field *Current #*.

### **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 afterward will receive increasing numbers from that value. The *Current#* field value will also be updated automatically.

## 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.

Methods for checking:


- The [Validate] button in the contract subtype form (see the sections "Card Contract Subtypes Form" and "Account and Device Contract Subtype ") 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. In this case, the *BIN Record* and *IPS Product* fields will not be 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 global parameter `IGNORE_IPS_PRODUCT` (see 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 Parm*s field.
- To disable the check for a specific subtype set the tag `IGNORE_IPS_PRODUCT`; in the subtype's *Add Parm*s field.

When the `IGNORE_IPS_PRODUCT` parameter is set (i.e. when checking of subtype fields is disabled), a Product will be approved (and financial institution checked) with the *BIN Record* and *IPS Product* fields empty or incorrectly filled in. In this case, a warning message will be generated (an error message not requiring correction).

 Note that when the `IGNORE_IPS_PRODUCT` parameter is set (when the *BIN Record* and *IPS Product* fields are empty or incorrectly filled in, WAY4 Datamart ETL processes will work incorrectly (processes for importing data to WAY4 Datamart), and "QMR" and "QOC" reports.

## Configuring Preferred Counterparties

Users may set additional conditions for operations in the WAY4 system according to preferred counterparty parameters.

The form "Preferred for <name of subtype>" (see Fig. 29 in section "Card Contract Subtypes Form") can be accessed by clicking on the [Preferred] button in the form "SubTypes for <name of card contract type>". For more details on working with preferred counterparties, see the Preferred Counterparties Administrator Manual).

## 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). For more information about message templates, see the "Configuring Message Templates" section of the "Configuration of Client Messages" document.

## 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 Fig. 30.

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

Code	Name	Offline PIN (fill only for Chip cards)	Contactless (fill only for Chip cards)	Handbook Code
E0001	E0001			Plastic Code
E0002	E0002			Plastic Code
E0003	E0003			Plastic Code
E0004	E0004			Plastic Code
L0001	L0001			Plastic Code
V0001	V0001			Plastic Code
V0002	V0002			Plastic Code
V0003	V0003			Plastic Code
V0004	V0004			Plastic Code
V0006	V0006			Plastic Code
V0007	V0007			Plastic Code

Fig. 30. "Plastic Codes" form

The 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.

## Appendix 1. 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 authorisation. For a simple contract (not a subcontract) the value of this parameter is calculated as the following:

$$\text{Own\_Amount\_Available} = \text{Current\_Balance} + \text{Credit\_Limit} - \text{Blocked\_Amount}$$

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

### Check, See Main

Amount\_Available calculation method when the *Auth Scenario* parameter is set to "Check":

$$\text{Amount\_Available} = \min(\text{Own\_Amount\_Available}, \text{Main\_Amount\_Available}),$$

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":

$$\text{Amount\_Available} = \text{Main\_Amount\_Available},$$

where *Main\_Amount\_Available* = *Amount\_Available* of the main contract, calculated in the same way.

To illustrate through a specific example:



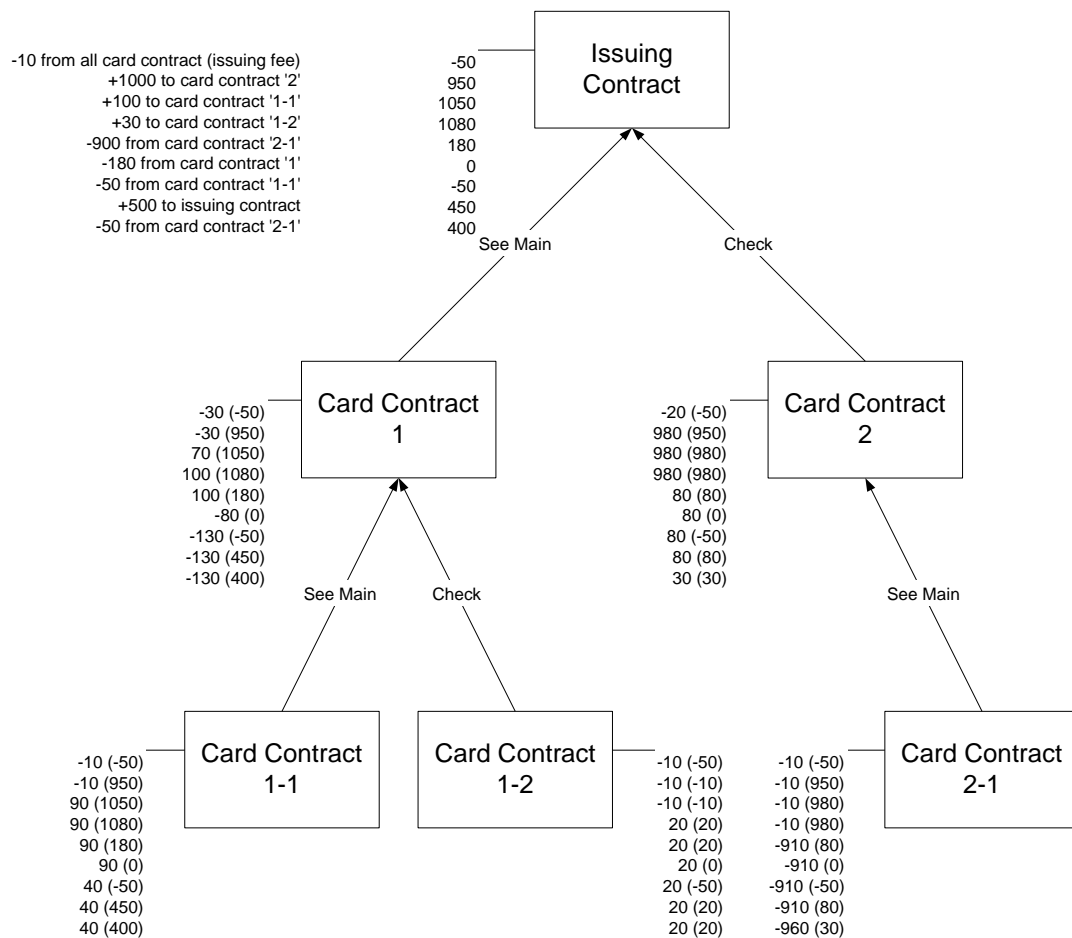


Fig. 31. Changes in a contract's amount available through various authorization scenarios

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 euros (EUR) was charged from every card contract. The account *Card Contract 1* is the consolidation of accounts *Card Contracts 1-1* and *Card Contract 1-2*. So fund transfer from these sub-accounts 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 is reflected in 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* contract 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 *Card Contract 2* is equal to the lesser of 980 and 1080, 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 *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* contract is equal to the lesser of 20 and 1080, that is, 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 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* (180); the amount available of *Card Contract 1-1* is equal to the amount available of its main contract (180); the amount available of the *Card Contract 1-2* contract 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, 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 *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 *Card Contract 1-2* is equal to the lesser of 20 and 0, that is, 0 EUR.
- 50 EUR were charged to the 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 *Issuing Contract* is equal to -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* contract 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 the *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 *Card Contract 1* is equal to the amount available of its main contract (450); the amount available of the *Card Contract 1-2* contract is equal to the lesser of 20 and 450, that is, 20 EUR.
- 50 EUR were charged to the card contract account *Card Contract 2-1*. This transfer was reflected in the consolidated contract account *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 U.E. 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 enacted, at the closing of a billing cycle, the balance of the subcontract is zeroed out.

## 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.

## Appendix 2. Tags in Products and Contract Subtypes

When the `USE_TRANS_AMOUNT` and `AUTH_USE_DOMESTIC` parameters (tags) are used, note the priorities for setting them (see below, in descending order):

- Service – the tag in a Service has the highest priority and redefines all other setting for this tag.
- Transaction subtype. The value of the tag in the transaction subtype is inherited on the Service level (if there is no redefining setting in the Service itself). I.e. this setting determines rules for all Services that will be used for transactions with this subtype.
- Additional Package.
- Product.
- Service Package.
- Financial Institution.
- Global parameter of the same name.

Recommended schemes for redefining the `USE_TRANS_AMOUNT` and `AUTH_USE_DOMESTIC` tags:

- If different rules are used (set with these tags) for different financial institutions or Products, a tag is redefined on the financial institution, Service Package, Product and Additional Package level (according to increasing priority).
- If the same rules are used (set with these tags) for all institutions/Products but these rules may differ for different transaction types, tags should be redefined on the level of separate transaction subtypes. A tag set in a transaction subtype is redefined on the level of separate Services.
- If different rules are used (set with these tags) in different institutions/Products and for different transaction types, these rules should be defined directly on the Service level.



These parameters (tags) are not redefined on the tariff level since the tags are processed when processing a document and in terms of system performance, it's not efficient to search for a tariff.

Name	Value	Description
Tags in the <i>Custom Data</i> field of a Product		
NO_COPY		<p>When copying selected parameters of a template financial institution, makes it possible for the Product of a child financial institution to not copy the Accounting Scheme, Service Package, Report Type, etc.</p> <p>When synchronising a Product Template with a related Product in another financial institution, make it possible to not synchronise the parameters of a child Product in a hierarchy. To do so, the tag must be set in the parameters of the related child Product.</p>
FREE_SUBTYPE		<p>When this parameter is set, the contract subtype is not inherited from the Product based on which the contract is registered. This allows the subtype to be redefined for a contract (by default, the subtype of a contract created with a Product cannot be changed without changing the subtype for the Product).</p> <p>When approving the Product:</p> <ul style="list-style-type: none"> <li>• In "Check Contract" mode, a check is made for the correspondence of the Product subtype and subtype in the contract (the contract number must correspond to the range specified in the subtype in the Product). If the contract's subtype differs from the subtype in the Product: <ul style="list-style-type: none"> <li>♦ if the FREE_SUBTYPE tag is set, an error related to the non-correspondence of the subtype in the Product and contract is not returned and the Product is approved.</li> <li>♦ If the FREE_SUBTYPE tag is not set, an error is returned and the Product is not approved.</li> </ul> </li> <li>• In "Check and Apply" mode, a check is made for the correspondence of the Product subtype and subtype in the contract (the contract number must correspond to the range specified in the subtype in the Product) and changes to existing contracts are applied: <ul style="list-style-type: none"> <li>♦ if the FREE_SUBTYPE tag is set the Product is approved without the contract inheriting the subtype from the Product.</li> <li>♦ if the FREE_SUBTYPE tag is not set, if the contract number falls in the range of the new subtype in the Product, the contract inherits the subtype from the Product. Otherwise, an error is returned that the Product is not approved.</li> </ul> </li> </ul>

Name	Value	Description
PARENT_LIMIT_PART	<tag name>	<p>This tag makes it possible to specify a credit limit for cards with "Check" and "Billing Limit" authorization scenarios as a percentage of the credit limit on the parent account contract.</p> <p>This tag is set in the Product of the card contract and specifies the tag name (arbitrary name) with which the percentage of the credit limit is specified. The percentage can be set:</p> <ul style="list-style-type: none"> <li>• In the EXT_DATA field of a contract.</li> <li>• In the <i>Custom Data</i> (CUSTOM_DATA) field of a Product.</li> </ul> <p>It is assumed that setup in the Product is the primary setup. Further changes to the percentage should be made in the contract.</p> <p>If the PARENT_LIMIT_PART tag is set, but a percentage is not specified in the Product or contract, the available credit limit for these cards will be "0".</p> <p>Attention! When the PARENT_LIMIT_PART tag is set a fixed credit limit cannot be set for a card. In this case an error will occur.</p>
USE_AUTH_FX	"Y" "N"	<p>This tag is used to redefine the value of the global parameter of the same name. The parameter specifies the date for the FX rate used when processing financial documents. When the "Y" value is set, the FX rate on the date of authorisation will be used when processing a financial document. For more details, see the document "WAY4™ Global Parameters".</p>
AUTH_USE_DOMESTIC	"Y" "C" "A" "N"	<p>The tag can be used to redefine the value of the tag of the same name set on the Service Package, financial institution, transaction subtype level, as well as the the value of the global parameter AUTH_USE_DOMESTIC (for more details, see the document "WAY4™ Global Parameters").</p>
USE_TRANS_AMOUNT	"Y" "N" "F"	<p>Used to determine the method for calculating the amount debited from "our" card accounts when a financial document imported from payment systems is posted, as well as the amount blocked when processing an authorization request:</p> <ul style="list-style-type: none"> <li>• "F" value – the transaction amount is blocked and debited from the contract if the card has an account in the transaction currency.</li> <li>• "Y" value – the transaction amount is always blocked and debited from the contract.</li> <li>• If the tag is not set or its value is "N", the settlement amount will be debited from the card.</li> </ul> <p>The tag redefines the tag of the same name set on the Service Package and financial institution level, and the global parameter of the same name.</p>

Name	Value	Description
		For more information, see the section "Calculating the Debited Amount ("USE_TRANS_AMOUNT" tag) of the document "Financial Institutions".
<balance type code>_CYCLE	<number of cycles> <custom parameter code>	Allows historical balance type values to be totalled for a specified number of cycles (redefines the CYCLE=<number of cycles>; tag set in the balance type). The obtained value can be used for further analysis of correspondence with set ranges (ranges are set using tariffs with the Service Limit role) and for setting the corresponding classifier. For details, contact WAY4 vendor representatives. The tariff module is supplied according to an additional agreement with the WAY4 vendor.
USE_AUTO_STORNO	"A" "Y" "R"	This tag is used together with the global parameter USE_AUTO_STORNO in the Reversal Management module. For more information, see the documents Reversal Management, Reversal Management Limited. The Reversal Management module is not included in the WAY4 basic configuration and is supplied according to a separate agreement with the WAY4 vendor.
SKIP_FOR_DUPLICATE		The SKIP_FOR_DUPLICATE; tag makes it possible to not inherit some subordinate contracts when creating a copy (Duplicate) of a contract, for example of a card when reissuing a lost or stolen card. To enable this option, set the appropriate subordinate contract in Product parameters.
SAVE_PAYMENT_FROM	<account type code>	<p>If the "Billing Limit" value of the Auth Scenario parameter is set in the card contract, by default, this card contract's balance is reset at the end of the billing cycle (both for a negative, or for a positive balance on the card). When the tag SAVE_PAYMENT_FROM=P; is specified (where "P" is the CI Deposit account type code):</p> <ul style="list-style-type: none"> <li>• If a positive balance remains on the card at the end of the billing cycle (balances are analysed for accounts with the "Yes" value of the <i>Is Am Av</i> parameter), the balance is not reset, is shown on the card's CI DEPOSIT account (account set with the tag SAVE_PAYMENT_FROM) and is used to calculate the card contract's amount available in the next billing cycle.</li> <li>• If the card balance at the end of the billing cycle is negative, the card balance is reset.</li> </ul> <p>The tag SAVE_PAYMENT_FROM=&lt;account type code&gt; is set in the <i>Custom Data</i> field in the Product for the main account contract.</p>

Name	Value	Description
PAYMENT_RESTORE	"ALL"	For a card contract payment to be reflected in limiters with the same code of other card contracts (all these card contracts must have the same parent contract), set the PAYMENT_RESTORE tag with the "ALL" value in the <i>Custom Data</i> field of this child Product. The payment will be shown in all other cards after debt is paid for the current card contract. Used when setting up a limiter (limit) to issue cash by credit card (see the section "Limiting Cash Withdrawal with a Credit Card (Configuring a Credit Limit for Cash Withdrawal Transactions)" of the document "Usage Limiters").
SYNC_PERSONAL_TARIFFS	"N"	The SYNC_PERSONAL_TARIFFS=N; tag redefines the "Y" value of the global parameter SYNC_PERSONAL_TARIFFS=N. The parameter is used in synchronising template tariff data and the data of personal tariffs created earlier on the basis of these templates (see the document "WAY4™ Advanced Tariff Management"). Synchronisation is not performed when the value is "N".
ATTACH_DEF_FROM_SUBS		When approving a contract, a check is made for contract custom parameter values set in the ATTACH_BY_CODE list in the Product. If the corresponding parameter is not set for the contract, the default value from the Product will be set. If the ATTACH_DEF_FROM_SUBS tag is set in the Product, a search is made and default values are inherited from subcontracts. If a default value is already set for the parent contract, values are not inherited from subcontracts.
INST_NUM_S	<options for the number of instalments, separated by commas>	The tag is used in the WAY4 Instalments module to simulate instalment plans. The tag makes it possible in the Product to set options for the number of instalments for which plans in the "Simulated" status will be created. See the section "Configuring Instalment Plan Simulation" in the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
SHIFT_NUM_S	<options for the number of periods by which an instalment plan is shifted, separated by commas>	The tag is used in the WAY4 Instalments module to simulate instalment plans. The tag makes it possible in the Product to set options for the number of periods by which an instalment plan is shifted, for which plans in the "Simulated" status will be created. See the section "Configuring Instalment Plan Simulation" in the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
INST_ADV_FEE_WAIVE_ON_CLOSE	"Y" "N"	The tag is used in the WAY4 Instalments module to set up rules for working with "Advance Fee" fees (fee from the start of the current billing cycle to the payment date) when closing an active plan.



Name	Value	Description
		<p>If the "Advance Fee" must be waived when closing a plan, set the tag value to "Y".</p> <p>The tag redefines the same global parameter.</p> <p>The module is supplied according to an additional agreement with the WAY4 vendor.</p>
INST_MIN_AUTO_ER_AMOUNT	<amount>	<p>The tag is used in the WAY4 Instalments module.</p> <p>In automatic early repayment, it is possible to set a limit for the minimum amount available required for automatic repayment. I.e. repayment is made if the amount available is equal to or more than the set amount. The minimum amount is set as the value of the INST_MIN_AUTO_ER_AMOUNT tag.</p> <p>The module is supplied according to an additional agreement with the WAY4 vendor.</p>
AUTO_ER_KEEP	"T" "P"	<p>The tag is used in the WAY4 Instalments module. In automatic partial early repayment a new plan can be calculated keeping the number of instalments or the size of an instalment from the original instalment scheme. To do so, set the AUTO_ER_KEEP tag with one of the following values:</p> <ul style="list-style-type: none"> <li>▪ "T" – to keep the number of instalment periods.</li> <li>▪ "P" – to keep the amount of an instalment in the instalment plan.</li> </ul> <p>The module is supplied according to an additional agreement with the WAY4 vendor.</p>
STOCK		<p>If secondary (normalisation) processes do not have to be executed for a specific large hierarchy of contracts (bank contracts or specific issuing contracts), for example, for a hierarchy of cards that have been issued but not yet sold. When issued, these cards are put in a special technical contract and then transferred to the bank's balance, i.e. put into real contracts (using applications, see the document "Advanced Applications R2"). In this case, after setting the STOCK; tag in the <i>Custom Data</i> field of a Product for a top-level technical contract, the corresponding data in the contract hierarchy are not updated/inherited when contracts are approved. Note that this tag cannot be used for usual issuing/acquiring contracts.</p> <p>The following process are secondary (normalisation) processes that can be disabled with the STOCK; tag:</p> <ul style="list-style-type: none"> <li>• Inheritance of preferred counterparties down the contract hierarchy</li> <li>• Definition of relations between contracts in a hierarchy for the "Extra Limit Normalization" process.</li> </ul>

Name	Value	Description
		<ul style="list-style-type: none"> <li>Filling in the CONTRACT_LEVEL field in the ACNT_CONTRACT table to show contracts in a certain order in the interface (contracts of the same level).</li> </ul>
STOP_PARM_INH_UPP	<list of contract custom parameters separated by commas>	Set in a card contract's Product and prohibits inheritance of the specified parameters from cards with this Product to a higher-ranking account contract (main contract in a "Main/Sub" hierarchy). Can be used, for example, to disable inheritance of a parameter from Supplementary cards. The tag can be set in a certain card contract.
STOP_PARM_INH	<list of contract custom parameters separated by commas>	Set in the Product of an account contract (main contract in a "Main/Sub" hierarchy) and prohibits inheritance of the specified parameters to an account contract (contracts) created for this Product from all card contract subordinate to it.
INST_PAYIN	"I" "B"	<p>The INST_PAYIN tag redefines the same global parameter. The tag is used when generating a response to an authorisation request from a terminal that supports MasterCard Instalments. The tag is used in instalment plan simulation mode.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>"I" – only generation of an instalment plan is possible (default value)</li> <li>"B" – standard payment and generation of an instalment plan is possible.</li> </ul> <p>See the section "Configuring Functionality to Pay for Transactions using Simulated Instalment Plans" of the document "Instalment Loans in WAY4".</p>
MIN_BILLING	<minimum billing cycle length in days> "C"	The parameter makes it possible to define the minimum billing cycle. The value is set in days. If the length of a billing cycle for a new contract is less than the value of this parameter, the end date of the next billing cycle will be used as the end date of the first billing cycle. The MIN_BILLING=C; tag specifies that there may be only one Billing Date in a calendar month. If this condition is not met, the end date of the billing cycle is shifted to the end date of the next billing cycle.
FIRST_BILLING	<number of days> ANY <number of months>	<p>"The value of the FIRST_BILLING tag redefines the value of the MIN_BILLING tag when calculating the end date of the first billing cycle. The FIRST_BILLING=ANY; tag makes it possible to set any length for the first billing cycle.</p> <p>The FIRST_BILLING=&lt;number of days&gt;; tag makes it possible to set the minimum length of the first billing cycle, in days.</p> <p>FIRST_BILLING=&lt;number of months&gt;;FIRST_BILLING_UNIT=M; tags make it possible to set the minimum length of the first billing cycle in months.</p>

Name	Value	Description
FIRST_BILLING_UNIT	M	FIRST_BILLING=<number of months>;FIRST_BILLING_UNIT=M; tags make it possible to set the minimum length of the first billing cycle in months.
MAX_BILLING	<maximum billing cycle length in days>	The tag sets the maximum length of a billing cycle. The tag is only checked when recalculating the current billing cycle's end date. If the length of the current billing cycle must be increase when changing the billing cycle's end date, the MAX_BILLING tag is checked. A check is made that the length of the billing cycle does not exceed the length in days specified as the value of the MAX_BILLING tag. If it does, the current billing cycle remains unchanged and the new Billing Date is only applied in the next billing cycle.
MAX_BILLING_UNIT		The tag is used if the length of a billing cycle may not be more than a month. The tag is used together with the MAX_BILLING tag. The tags are only checked when recalculating the current billing cycle's end date. When MAX_BILLING_UNIT=M; is set if the length of the first billing cycle for a new contract is more than a month, the date will not be recalculated and the current billing cycle will remain unchanged.
MIN_BILLING_UNIT	"M"	The parameter is used if a billing cycle's length cannot be less than a month. Used together with the MIN_BILLING parameter. When MIN_BILLING_UNIT=M; if the length of a new contract's first billing cycle is less than a month, the end date of the next billing cycle will be used as the end date of the first billing cycle.
REPLACE_TO	<code of the new Product>	The tag is used when a card's Product must be changed when the card is reissued. See the section "Changing a Product when Reissuing a Card."
STORNO_LIAB		The tag is used for mapping Liability Products. For contracts created on the basis of these products, correction operations with Reversal Management tools can be performed (both manually or automatically). When the transaction correction procedure is run (or procedures for reviewing a contract's lifecycle from a certain date) for a sub contract in a "Main/Sub" hierarchy, if the main contract in the "Main/Sub" hierarchy has a contract with a "Liability" relation type, a check for the STORNO_LIAB tag in the Product is made for it. When the tag is present: <ul style="list-style-type: none"> <li>• A plan is generated for the top contract in the "Liability" hierarchy with the STORNO_LIAB tag configured (i.e. there are other higher-ranking contracts for a Liability contract, a check for the STORNO_LIAB tag in the Product is made for them also).</li> </ul>

Name	Value	Description
		<p>• This plan is general and includes corrections for all subordinate contracts. Both for Liability contracts (if there are corrections) and for all subordinate contracts in the "Main/Sub" hierarchy (i.e. for all Main/Sub branches in the hierarchy). If correction is run for the Top contract for which subordinate Liability contracts are created, a plan for the Top contract is only generated when STORNO_LIAB is set up.</p> <p>If a Main contract (main contract in a Main/Sub hierarchy) is also the Top contract (subordinate Liability contracts are set up for it), it is mandatory for the STORNO_LIAB tag to be in its Product, since in this case corrections must be made for the entire combined (Main-Sub-Liability) tree. For more information, see the documents Reversal Management, Reversal Management Limited.</p> <p>The Reversal Management module is not included in the WAY4 basic configuration and is supplied according to a separate agreement with the WAY4 vendor.</p>
CR_LIMIT_TRF	<code>	<p>Used to determine the lower and upper boundaries of a credit limit, as well as the default credit limit, using tariffs with the "Threshold" role. For Products whose limits must be defined by tariffs, the CR_LIMIT_TRF=&lt;code&gt;; tag must be set. A search will be made for tariffs with the MIN_&lt;code&gt;,MAX_&lt;code&gt;,DEF_&lt;code&gt; codes.</p> <p>If the CR_LIMIT_TRF tag is set in the Product, limit boundaries (and the default value) set in the Product will not be used, even if the tariff is not found or is equal to zero. When searching for tariffs, if a tariff value is not found, the corresponding value of the limit is considered as being equal to zero (i.e. interpreted as no limits).</p> <p>For more information, see the section "Contract Credit Limit" of the document "WAY4 Advanced Tariff Management".</p>
Tags in the <i>Add Parms</i> field of a contract subtype:		
NEW	<Production event code>	<p>This parameter specifies the PRODUCTION_EVENT from which parameters for marking for issue will be taken. The tag value is the value of the <i>Code</i> field of the form "Full → Configuration Setup → Transaction Types → Production Events".</p>

Name	Value	Description
RENEW	<Production event code>	This parameter specifies the PRODUCTION_EVENT from which parameters for marking for reissue will be taken. The tag value is the value of the Code field of the form "Full → Configuration Setup → Transaction Types → Production Events".
MIN_CARD_EXPIRE	<number of months>	Makes it possible to redefine the corresponding global parameter MIN_CARD_EXPIRE – the minimum card expiry date. By default, this is 3 months.
MAX_CARD_EXPIRE	<number of months>	Makes it possible to redefine the corresponding global parameter MAX_CARD_EXPIRE – the maximum card expiry date. By default, this is 60 months.
PIN_SET	"Y"	This parameter with the "Y" value makes it possible to change the PIN without requesting the previous PIN.
EXPIRE_DATE_FROM_MAIN	"Y" "N"	When this tag is set with the "N" value, "Expire Date" for an applet is not inherited from the card to which the applet is linked. If the tag is not set or has the "Y" value, the applet's "Expire Date" is specified from the card.
DAYS_TO_RENEW	<number of working banking days>	DAYS_TO_RENEW=<number of working banking days>; makes it possible to configure automatic card reissue. A card contract for reissue is marked in advance by the number of days specified in the tag until the card's expiry date + 1 day. To decrease the load on production, the RENEW_DAYS_RANGE=<number of calendar days>; is additionally used.
RENEW_DAYS_RANGE	<number of calendar days>	Works together with DAYS_TO_RENEW. Allows distribution of the card marking process for scheduled reissue over a specified number of days, to avoid peak load on the production subsystem. Allows distribution of the card marking process for scheduled reissue over a specified number of days until the date of scheduled reissue defined by the DAYS_TO_RENEW tag. I.e. when distributing the marking process, the number of days set by the RENEW_DAYS_RANGE parameter is added to the number of days specified in the DAYS_TO_RENEW tag.
CARD_NAME_LENGTH	<positive integer> whole	Tag can be used to redefine the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters").
COMPANY_NAME_LENGTH	<positive integer> whole	Tag can be used to redefine the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters").

Name	Value	Description
SHIFT_CARD_EXPIRE_FROM	<whole integer from 0 to 99>	Tag can be used to redefine the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters").
CARD_RENEW_ADVANCE	<from 0 to 12>	Tag can be used to redefine the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters").
CARD_RENEW_FROM_TODAY	<values are from -12 to 12>	Tag can be used to redefine the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters").
SEND_PVV_2_PS	"N"	The SEND_PVV_2_PS=N; tag redefines the global parameter SEND_PVV_2_PS with the "Y" value. The SEND_PVV_2_PS=N; tag disables sending PVV data to the payment system when changing a PIN.
IGNORE_IPS_PRODUCT		The IGNORE_IPS_PRODUCT; tag redefines the global parameter of the same name. For more information, see the section "IGNORE_IPS_PRODUCT" of the document "WAY4 Global Parameters".
PAN	MAIN	The "PAN=MAIN;" tag means that the card number for the applet is inherited from the card's main financial application. See the document "Configuring WAY4™ for Smart Card Issuing".
SEQV_N_MASK	9_	The SEQV_N_MASK=9_; tag that contains a mask of the plastic number makes it possible to process several numbers (SEQV_NUMBER) of additional card applications for "MIR" payment system cards. The PAN=MAIN;SEQV_N_MASK=9_; tag is mandatory in additional card application (Applets) card subtypes when setting up a hierarchy for "Mir" payment system cards. See the document "Configuring WAY4™ for Smart Card Issuing".
Tags in the <i>Validation Type</i> field of a contract subtype:		
CHECK_RANGE	"Y" "N"	The "Y" value of the CHECK_RANGE parameter allows enabling a check of the correspondence of the contract number to the range of numbers set on the contract sub-type level (the <i>BIN</i> , <i>Min#</i> , <i>Max#</i> fields). The check is executed for both card and account contracts. By default (if the parameter is not set or set to "N"), a check is made only for the card contract. The parameter is used to check numbers of account contracts entered manually.

Name	Value	Description
INI_STATE	<Contract Code>      Status	By default in registration and successful approval of a contract, it is assigned the "Account OK" status. The parameter "INI_STATE" allows an initial status to be set for a contract with this subtype that differs from "Account OK" The contract status code is specified as the value (the <i>Code</i> field of the form "Full → Configuration Setup → Contract Types → Contract Statuses").
INITIAL_LOCK		This parameter makes it possible to block new plastic from the time of card production by the PIN Management module until the plastic is issued to the client in order to avoid unauthorised use of reissued bankcards.
U_PBT;		The U_PBT; parameter makes it possible to automatically unblock new plastic on the first successful PIN transaction (if the card was blocked using the INITIAL_LOCK parameter.
BLANK_CARD_NAME		For VISA Virtual cards, this parameter makes it possible to process embossing fields in the contract in the following way: <ul style="list-style-type: none"> <li>• If the fields are not filled in, when issuing a card nothing is embossed on the plastic and nothing is specified in the "Cardholder Name" parameter in the <i>Track 1</i> field. The value "VISA ELECTRON CARDHOLDER" is specified on the card by embosser settings.</li> <li>• If the fields are filled in, when a card is issued, the client's first and last name will be embossed on the plastic and coded in the "Cardholder Name" parameter in the <i>Track 1</i> field .</li> </ul>
CVV_ONLY		This parameter allows a card to be issued without a PIN (VISA Virtual). It is possible to used the standard production events (Production event - "New Card", Card Event - "Produce Card", Production Type - "Replace All").
CLOSE_PREV_PLASTIC		When reissuing a card with this parameter in the subtype all previous plastics will be closed and the corresponding number of Events with the predefined code CLOSE_PLASTIC_TRANS_STATUS will be opened for the card. Plastic is closed when information about card production is received (when data is imported from PIN Management).
UNLOCK_BY_TC	<list of transaction condition codes, separated by commas>	The UNLOCK_BY_TC; tag makes it possible to specify transaction conditions for which a new plastic must be unlocked. Transaction condition codes (value of the CODE field in the TRANS_COND table), separated by commas are used as the value. It is recommended to specify transaction group conditions. The tag is used together with the CLOSE_PREV_PLASTIC_BY_TC tag.

Name	Value	Description
CLOSE_PREV_PLASTIC_BY_TC	<list of transaction condition codes, separated by commas>	The tag makes it possible to specify transaction conditions for which an old plastic must be locked. Transaction condition codes (value of the CODE field in the TRANS_COND table), separated by commas are used as the value. It is recommended to specify transaction group conditions. The tag is used together with the UNLOCK_BY_TC; tag.