WAY4™ Service Packages

Contents

Introduction

This document is intended for bank or processing centre employees responsible for configuring WAY4. It contains information on the registration and configuration of new products.

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

- "Documents"

- "WAY4™ Dictionaries"

- "Events"

- "Issuing Module"

- "Acquiring Module"

- "Products and Contract Subtypes"

- "WAY4™ Global Parameters"

- "Usage Limiters"

- "Preferred Counterparties"

- "Balance Types"

- "WAY4™ Advanced Tariff Management"

- "Risk Monitoring"; the WAY4 Real-Time Risk Management module

- "Instalment Loans in WAY4™"

The following conventions are used throughout the document:

- Field names in screen forms are displayed in *italics*.

- Names of screen form buttons are shown in square brackets, such as [Approve].

- The sequence for selecting a menu item is given using arrows, as in "Configuration Setup → Contract Types".

- Warnings that there is a risk of making an incorrect action are marked with the ⚠ sign.

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)

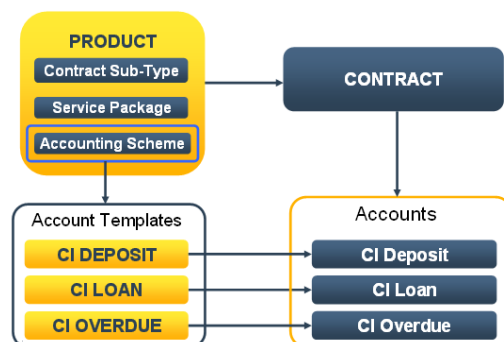

Fig. 1. Relations between system objects

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. Configuring Service Packages

In WAY4™, a Service is a rule for processing operations.

Examples of operations:

- Making a purchase using a card at a payment terminal (POS)

- Issuing cash at an ATM

- Request for a mini-statement

- Charging an annual fee for servicing a card

Main rules for processing operations:

- Permitting or forbidding an operation

- Choice of contract account type for generating movement of funds

- Rules for charging fees (rules for calculating fee amounts, fee direction, etc)

Services are combined in Service Packages selected when registering Products or contracts. By doing so, a set of rules is specified for a Product or contract according to which operations configured for this Product (contract) will be processed.

WAY4 is delivered with a fixed set of Service Packages that can be configured according to the bank's current needs. When creating new Products, bank employees add the corresponding Service Packages.

Various screen forms can be used for working with Service Packages in WAY4. Screen forms provide access to the parameters of Packages and their Services.

"Service Packs" Form

To configure a new Service Package, in the user menu select the item "Full → Configuration Setup → Products → Service Packs" to open the "Service Packs" grid form (see Fig. 2) that contains all registered Service Packages.

ⓘ Service Packages can also be configured in special forms (see the section "Special Forms for Working with Service Packages").

	Category	Institution	Client	Contract Type	Name	Code	For Contracts	Parent Pack	Fee Contract	Tariff Domain	Is Ready	Special Parms	
Service Packs								‹‹ ‹ › ››		1 of 1188		✕
	Account	Principal	Accountant	Bank Account	001-RBS Bank Account		Product				Ready		
	Account	Principal	Accountant	Bank Account	001-RBS Teller		Product				Ready		
	Account	Principal	Accountant	Bank Account	001-Teller		Product				Ready		
	Account	Principal	Accountant	Nostro	001-Affiliated Nostro		Product				Ready		
Ins	Del	Query	Approve	Misc	Source	Target	Additional	Usage	Events	Messages			

Fig. 2. Form for entering information about a new Service Package

A new Service Package can be entered in the database in either of two ways:

- By inserting an empty row by clicking the [Ins] button and filling in the fields.

- By copying an existing Service Package and changing the values of fields in the form according to the requirements of the new Package (see the section "Copying Service Packages").

The "Service Packs" form contains fields for setting the main parameters of Service Packages. For a description of the fields, see the section "Main Parameters of Service Packages".

The form also contains the following navigation buttons:

- The [Approve] button contains the following context menu items:

  - [Approve] – approve Service Package changes in the database. For more information, see the section "Approving Service Packages".

  - [Duplicate] – copy a Service Package (see the section "Copying Service Packages").

  - [Check] – check Service Package Parameters (see the section "Checking Service Packages").

- The [Details] button is used to access the settings of additional parameters of Service Packages (see the section "Additional Parameters of Service Packages").

- The buttons [Misc], [Source], [Target] are used to access the forms with settings of various service types (for more information, see the section "Service Configuration").

- The [Additional] button is used to configure additional Service Packages (see the section "Configuring Additional Service Packages").

- The [Usage] button is used to configure Service Package usage limiter templates (see the section "Configuring Usage Limiter Templates").

- The [Events] button is used to configure Events to automatically activate/deactivate additional Service Packages (see the section "Configuring Events").

- The [Group Msg] button is used to configure client message templates (see the section "Configuring Message Templates (Group Msg))".

- The [Preferred] button is used to configure preferred counterparties (see the section "Configuring Preferred Counterparties (Preferred)").

Main Parameters of Service Packages

Category

The *Category* field determines the contract category (Card, Account, Device) for which the Service Package will be used.

Institution

The Institution field specifies the financial institution whose contract can use the Service Package.

Client

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

Contract Type

The *Contract Type* field shows the type of contract for which the given Service Package is used.

Name

The *Name* field specifies the name of the Service Package which will be used for registering contracts. The following name format is recommended: "NNN-name", where NNN is the last three digits of the number of the financial institution indicated in the *Branch Code* field of the "Financial Institutions form (Full → Configuration Setup → Main Tables → Financial Institutions).

Code

The *Code* field contains a special code used when importing files from external sources to associate the imported data with a specific Service Package.

For Contracts

The *For Contracts* field is used to separate all Service Packages into the following groups:

- "Product" – Service Packages used for creating Products and registering contracts (when the "Product" value is selected in this field).

- "Additional" – additional Service Packages to be linked to main Service Packages to expand Product or contract functionality (for more information, see the section "Configuring Additional Service Packages").

  Can be used to group Services with the ATTACHMENT tag. In this case, the Package is used to attach/detach an additional fee to all the Package's main Services. The tag is described in the section "Tags Used when Configuring Service Packages and Services"

- "Risk Rules" – special Service Packages used for risk control. These Service Packages are a variation on additional Service Packages and are also used in Product configuration and contract registration (see the Risk Monitoring User Manual; the WAY4 Real-Time Risk Management module.)

This field is used to avoid mistakes where an Additional Service Package is assigned to a Product or contract instead of a regular Service Package.

Parent Pack

To make it easier to synchronise parameter changes in Service Packages belonging to different financial institutions, it is possible to make changes to one Service Package with the subsequent automatic change in tariffs of the other Service Packages linked to it. The *Parent Pack* field is used to link Service Packages with the main Service Package determining their tariffs.

After approval in the database of changes to the "parent" Service Package, this information is automatically reflected in all other Service Packages that refer to the given Package in the *Parent Pack* field.

When parameters of the main Service Package are inherited, all settings of the "parent" package are copied, including additional Service Packages, etc. For Events, message templates are also inherited from the Events of the "parent" package.

A subordinate Service Package cannot have the value "Ready" in the *Is Ready* field if the Service Package referred to in the *Parent Pack* field has the status "Not Ready".

The *Parent Pack* field of a financial institution's Service Package is filled in from a drop-down list containing the Service Packages registered in the system which are used for the same type of contract (*Contract Type* field) as the given Service Package.

⚠ *Parent Pack* and *Base Pack* configurations are not intended for use in one hierarchy of Service Packages (for example, in the *Base Pack* field of Service Package 1, Service Package 2 is specified, and in the *Parent* field of Service Package 2, Service Package 3 is specified). Such configurations may cause system failure.

Fee Contract

The *Fee Contract* field contains the contract whose accounts will be used for the transfer of fees.

Tariff Domain

The *Tariff Domain* field for each Service Package may contain one of the tariff domains registered in the system. If the bank's distribution package does not include the WAY4 Tariffs module, the field is empty.

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

Is Ready

The *Is Ready* field shows whether changes to the given Service Package were approved.

⚠ If an Additional Service Package is configured for the given Service Package, after approval of changes in the Additional Service Package, the main Service Package and all its subordinate Service Packages will have the status "Not Ready".

When changes are made to a Service Package for which contracts were earlier registered in the database, contract properties will be changed according to the value of the global parameter APPROVE_IMMEDIATE (see the WAY4™ Global Parameters User Manual).

Additional Parameters of Service Packages

To access additional information about a Service Package, use the form (see Fig. 3) opened by clicking the [Details] button in the Service Packages grid form.

Fig. 3. Form with additional information about a Service Package

This form contains fields which do not appear in higher-ranking Service Package forms (see the description in the sections below).

The form with additional information about a Service Package contains the following navigation buttons:

- The [Actions] button contains the following context menu items:

  - [Check] – checks the Service Package parameters (see the section "Checking Service Packages").

  - [Duplicate] – copies the Service Package (see the section "Copying Service Packages").

  - [Save Tagged Data] – saves tags entered in the *Special Parms* field (see the section "Entering and Editing Tags").

- [Approved] – opens the grid form with the current values of Service parameters (see the section "Current Service Parameters (Approved Services)").

- [Used by] – opens the grid form specifying Service Packages to which the given Service Package is linked, if it is an additional Service Package (see the section "Configuring Additional Service Packages").

- The [Tagged Data] button is used to optimise work with the *Special Parms* field. Clicking this button opens a grid form for entering and modifying tags in the *Special Parms* field (see the section "Entering and Editing Tags").

- The [UsageAppr] button is used to access the form containing records corresponding to all changes ever approved to all usage limiter template parameters of the given Service Package.

- The [Service List] button is used for batch configuration of tariffs for Service Packages. Tariff types can be selected in the form opened by clicking this button if the delivery package includes the WAY4 Tariffs module. The WAY4 Tariffs module is not included in the basic WAY4 configuration and is supplied under a separate agreement with the WAY4 system vendor.

Use for Acnt

The *Use for Acnt* field is used to separate all Service Packages into Service Packages used to create Products and register contracts (the value "Product"), additional

Service Packages (the value "Additional") and Service Packages used for risk control (the value "Risk Rules") (see the section "For Contracts").

Base Pack

This field remains for backward compatibility.

Additional Service Packages functionality (see the section "Configuring Additional Service Packages") should be used instead of this field.

PIN Tries

This group of fields is used to configure the rules for working with contracts when an incorrect PIN code is entered:

- *On US PIN Resp* – this field contains the value of the response code returned during authorisations of "on-us" cards at "on-us" devices when the maximum allowed number of PIN attempts is exceeded. The maximum allowed number of such attempts is set in the *Def PIN Tries* field.

- *Ext PIN Resp* – this field contains the value of the response code returned during authorizations of "on-us" cards at "foreign" devices when the maximum allowed number of PIN attempts is exceeded. The maximum allowed number of such attempts is set in the *Def PIN Tries* field.

- *Def PIN Tries* – the value of the maximum allowed number of PIN attempts. When registering a new contract, this value is used as the default value. The counter for unsuccessful attempts resets to 0 when the correct PIN code is entered if the maximum number of attempts was not exceeded.

Statistical Diagrams

This group of fields is used to set the rules for gathering statistical information:

- *Individual* – this field contains a drop-down list of registered Statistics Schemes to specify the Scheme that will be used for gathering individual statistics for contracts using the given Service Package.

- *Add to Main Tag* – a flag that determines (when the value is "Yes") that statistics will be gathered for contracts ranking above the contracts using the given Service Package.

- *Group by Pack* – this field contains a drop-down list of registered Statistics Schemes to specify the Scheme that will be used for gathering consolidated statistics for all contracts using the given Service Package.

Risk Scheme

The *Risk Scheme* field indicates the Risk Scheme for magnetic stripe cards. This value will be assigned by default to contracts using the given Service Package (see the "Risk Management Reports Setup" document).

Chip Scheme

The *Chip Scheme* field specifies the smart card parameter scheme that will be assigned by default to contracts using the given Service Package. This parameter can be redefined on the contract subtype level (in the *Chip Scheme* field).

Special Parms

The *Special Parms* field is used to define special parameters of Service Packages as tags. For more details, see the section "Tags Used when Configuring Service Packages and Services".

Default Fee Acc

The *Default Fee Acc* field is used to specify the default account for transaction fees described in the Service Package. The value of this field is reflected in Services included in this Package.

Use Def Service

The *Use Def Service* field specifies how default accounts are used. These account types are set when configuring transaction subtypes (see the section "Transaction Subtypes" in the Documents Administrator Manual).

The value of the *Use Def Service* field determines when to use this default value to process documents for which no Service was found, and consequently, no posting account is determined:

- "None" – a document for which no Service was found is rejected.

- "For All" – the default account is used for all documents for which no Service was found.

  ⓘ Note that the *Use Def Service* parameter with the "For All" value can only be used for bank contract Service Packages.

  It is not recommended to use the *Use Def Service* parameter with the "For All" value for card contracts.

  For example, if this value is specified for a card contract Package, funds in the contract will be blocked regardless of the *Is Am Av* field value ("Yes" or "No") for the default account. I.e. regardless of the "No" value in the *Is Am Av* field for the "default" account when the Service is absent, funds will be blocked in the contract and shown in the Total Blocked field of the contract's "Balance for…" form.

- For Dispute – the default account is only used for dispute documents for which no Service was found.

Special Forms for Working with Service Packages

In most cases, special forms are used with a list of Service Packages selected by financial institution and client category:

- To enter and edit information on Service Packages for issuing contracts of private clients, use the following forms from the "Full → Configuration Setup → Products → Issuing Private Products" menu group:

  - For card contracts, use the form "Private Card Service Packs" (see Fig. 4).

  - For account contracts, use the form "Private Account Service Packs" from the same group.

- To enter and edit information on Service Packages for issuing corporate contracts, use the following forms from the "Full → Configuration Setup → Products → Issuing Corporate Products" menu group:

  ▪ For card contracts, use the form "Corporate Card Service Packs".

  ▪ For account contracts use the form "Corporate Account Service Packs".

- To enter and edit information on Service Packages for acquiring contracts, use the following forms from the "Full → Configuration Setup → Products → Acquiring Products" menu group:

  ▪ For device contracts, use the form "Device Service Packs".

  ▪ For corporate account contracts, use the form "Corporate Account Service Packs".

- To enter and edit information on Service Packages for bank contracts, use the form "Full → Configuration Setup → Accounting Setup → Bank Service Packs".

Name	Contract Type	Parent Pack	For Contracts	Use Default	Code	Fee Contract	Special Parms	Is Ready
001-Our Priv JCB	Our JCB Cards		Product	For Dispute	PIJ	001-CLIENT_FEE		Ready
001-Our Local Card	Our Local Card		Product	For Dispute	PIL	001-CLIENT_FEE		Ready
001-Our Priv Electron/Plus	Our Electron/Plus		Product	For Dispute	PIEP	001-CLIENT_FEE		Ready
001-Our Priv Cirrus/Maestro	Our Cirrus/Maestro		Product	For Dispute	PICM	001-CLIENT_FEE		Ready
001-Our Priv EC/MC	Our EuroCard/MasterCard		Product	For Dispute	PIEM	001-CLIENT_FEE		Not Ready

Fig. 4. Form for entering and editing information on Service Packages for private card contracts

Note that fields and buttons in the abovementioned forms may vary.

The [Tariffs] button (similar to the [Service List] button in the additional Package parameters form) is used for batch configuration of Service Package tariffs. Tariff types can be selected in the form opened by clicking this button if the delivery package includes the WAY4 Tariffs module. The WAY4 Tariffs module is not included in the basic WAY4 configuration and is supplied under a separate agreement with the WAY4 system vendor.

Service Configuration

Service Types

Services fall into two main groups; these are transaction Services (Target Services and Source Services) and non-transaction Services (Miscellaneous services).

Source Services

The [Source] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form") is used to open the "Source for <name of Service Package>" form (see Fig. 5). This form contains information on Services used for operations where the contract using the given Service Package will be the source of transaction information. These Services are usually described in Service Packages for devices.

Fig. 5. Form for viewing and entering data on source Services

This form contains the following fields, which are not accessible in the form with detailed information about a Service.

- The *Target Type* field indicates the contract type of the counterparty who is the target of transaction data.

- The *Transaction Type* field specifies the transaction type.

The form contains the following control buttons:

- [Duplicate] – used to copy a Service in a Service Package (see the section "Copying Service Packages").

- [Full Info] – used to access the form with full information about a Service (see the section "Full Information about a Service").

- [Custom Fee] – used to configure a custom fee calculated as a percentage of the transaction amount without the main fee (see the section "Custom Fee").

- [History] – opens a form with the current values of this Service's parameters (see the section "Current Service Parameters (Approved Services)").

Target Services

Clicking the [Target] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form") opens the "Target for <name of Service Package>" (see Fig. 6). This form contains data on Services determining the permissibility and conditions of operations in which the contract using the given Service Package will be the target of transaction information. Usually, these Services are used for card contracts.

Fig. 6. Form for viewing and entering data on target Services

The form contains the following fields, which are not accessible in the form with detailed information about a Service:

- The *Source Type* field indicates the contract type of the counterparty who is the source of transaction data.

- The *Transaction Type* field specifies the transaction type.

The form contains the following control buttons:

- [Duplicate] – used to copy a Service in a Service Package (see the section "Copying Service Packages").

- [Full Info] – used to access the form with full information about a Service (see the section "Full Information about a Service").

- [Custom Fee] – used to configure a custom fee calculated as a percentage of the transaction amount without the main fee (see the section "Custom Fee").

- [History] – opens a form with the current values of this Service's parameters (see the section "Current Service Parameters (Approved Services)").

Miscellaneous Services

Clicking the [Misc] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form") opens the "Misc for <name of Service> (see Fig. 7). This form contains data on Services that describe fees charged when certain events occur, for example, card issuing, mini-statement request, etc.

Fee Type	Fee Day	Fee Dir	Rate Type	FX Type	Fee Curr	Base	%	Min	Is Ready	Name	Fee Code	Account Type	Fee Account
Card Production	0	Debit	Middle		USD	10,00	0,00	0,00	Ready	Card Production		CI Deposit	Cash Fees Passive USD
Card Renew Expired	0	Debit	Middle		USD	10,00	0,00	0,00	Ready	Card Renew Expired		CI Deposit	Cash Fees Passive USD
Card Fee Yearly	0	Debit	Middle		USD	10,00	0,00	0,00	Ready	Card Fee Yearly		CI Deposit	Cash Fees Passive USD
Balance Inquiry Fee	0	Debit	Middle		USD	1,00	0,00	0,00	Ready	Balance Inquiry Fee		CI Deposit	Cash Fees Passive USD
Online Statement	0	Debit	Middle		USD	1,00	0,00	0,00	Ready	Online Statement		CI Deposit	Cash Fees Passive USD

Misc for 001-Our Priv VISA   1 of 15

[Ins] [Del] [Query]   [Service - Full] [Custom Fee]   [History]

Fig. 7. Form for viewing and entering information about miscellaneous Services

This form contains the *Fee Type* field, which is not accessible in the form with detailed information about a Service. In this field, the fee type is selected from a drop-down list of non-transaction fee types available for the contract category set for the given Service Package (the list is set in the "Full → Configuration Setup → Transaction Types → Fee Types" form).

The form contains the following control buttons:

- [Service - Full] – used to access the form with full information about a Service (see the section "Full Information about a Service").

- [Custom Fee] – used to configure a custom fee calculated as a percentage of the transaction amount without the main fee (see the section "Custom Fee").

- [History] – opens a form with the current values of this Service's parameters (see the section "Current Service Parameters (Approved Services)").

Full Information about a Service

By clicking on the [Full Info] button in the forms "Source" (see Fig. 5 in the section "Source Services") and "Target" (see Fig. 6 in the section "Target Services"), and by clicking the [Service-Full] button in the "Misc" form (see Fig. 7 in the section "Miscellaneous Services"), users can access the form containing full information on each individual Service (see Fig. 8). This form can be used for the complete configuration of a Service.

Fig. 8. Form for viewing and entering complete Service information

The *Service Allowed* field of this form can have one of the following values:

- "Always" – the Service allows transactions to be made.

- "Offline on Issuing" – the Service prohibits authorizations, but processes financial documents registered in the system without authorization according to Service rules using a dispute contract account.

- "Offline on Acquiring" – when an online authorization request to make a transaction on this Service is received from a device, a financial document for further processing the transaction offline is generated in the system (this Service is like the "Credit" operation in the MasterCard payment system). The authorization process is terminated.

- "Never" – the Service prohibits authorizations and processing of financial documents.

Priority is a numeric indicator of priority, used when searching for a Service. When WAY4 searches for a Service and several Services meeting search conditions are found, the Service with the higher value in this field will be selected (a Service with a higher priority will be selected regardless of its details – regardless of the number of criteria set in it).

The *Is Ready* field shows that the changes made to the given Service were approved.

The remaining fields of this form are divided into three groups:

- "Transaction Parameters"

- "Tariff"

- "Posting"

The form also contains the following buttons:

- The [Actions] button contains the following context menu items:

  - [Save Tags] – saves tags entered in the *Service Details* field (see the section "Entering and Editing Tags").

  - [Duplicate] – copies a Service (see the section "Copying Service Packages").

- [History] – opens the form with the current values of the given Service's parameters (see the section "Current Service Parameters (Approved Services)").

- [Custom Fee] – used to set additional fees (see the section "Custom Fee").

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

Transaction Parameters

This group of fields determines transaction parameters:

- The *Contra FI* field shows the counterparty's financial institution.

- The *Subtype* field determines the counterparty's contract subtype.

- The *Settl Currency* field shows the settlement currency.

- The *Trans Curr* field shows the transaction currency.

  A Service's Settl Curr and Trans Curr field values are compared with the values in the corresponding fields of the document (*Settlement Currency* and *Transaction Currency*). If the value of the USE_TRANS_AMOUNT tag is "Y" (see the description of the USE_TRANS_AMOUNT tag in the section "Tags Used when Configuring Service Packages and Services"), when searching for a Service, both fields of the Service (*Settl Curr* and *Trans Curr*) are compared with the document's *Transaction Currency* field.

- The *Condition* field shows the conditions for making the transaction; the "Transaction Conditions" system dictionary contains a list of all conditions registered in the system.

  When configuring Services only group conditions for transactions should be used (these records have the name "<name> (Group)" in the "Transaction Conditions" list).

- The *Max Amount* field defines the maximum allowed amount of the transaction.

- The *Min Amount* field defines the minimum allowed amount of the transaction.

- The *Max Amount Curr* indicates the currency in which limitations are set on the transaction amount.

  The *Max Amount Curr* field is filled in with data from the *Settl Curr* field automatically when approving a Service Package if even one of the following fields is filled in: *Floor Limit*, *Min Amount*, *Max Amount*. If the *Settl Curr* field is not filled in, an error message will be displayed.

- The *Expiry Period* field is used as follows: according to the transaction parameters (including the value in the *Condition* field), the system determines the Service for processing a transaction. Then the system compares the date of the transaction with the date the corresponding document was processed. If these dates differ by a value larger than that indicated in the *Expiry Period* field of the Service found, the system searches for another Service. A new Service is determined according to the value of the *Late Conditions* field of the "Transaction Conditions" dictionary. For more information about working with

this dictionary, see the section "Transaction Conditions" in the Documents Administrator Manual.

Note that the Service will not be replaced if the *Expiry Period* field value is "0".

The value of this field affects document processing for transaction types with the parameter Category = "When Available": unblocking amounts and document posting are only possible during the interval determined by the value of the *Expiry Period* parameter from the document creation date. If this interval is exceeded, the document will be declined and a corresponding message will be generated. For more information on processing this category of financial documents, see the section "Posting "When Available" Type Documents" in the Documents Administrator Manual.

- The *Preference* field shows the counterparty category (see the Preferred Counterparties Administrator Manual).

- *SIC Group* – this parameter allows the additional specification of a Service for processing an operation depending on the merchant SIC group (see the section ""SIC Group" Dictionary" in the WAY4™ Dictionaries Administrator Manual). The system can search for Services by the value of this field only in Service Packages for card contracts (a contract's *Contract Category* is set to "Card").

- *Transaction Type Extension* – this parameter allows the additional determination of a Service for processing an operation taking into account the correspondence of the value of the *Transaction Type Extension* field and the value of the document's source_fee_code field.

  This field is filled in from a list configured in the "Transaction Type Extensions" form (Full → Configuration Setup → Transaction Types → Transaction Type Extensions). See the section "Transaction Type Extensions" of the document "Documents".

Tariff

This group of fields describes transaction fees:

- The *Fee Dir* field (mandatory) determines the fee direction (from a drop-down list), where the value "Debit" means that the client is charged a fee for a transaction; "Credit", the client is paid a fee for a transaction and "None", no fee is calculated.

- The *Fee Curr* field shows the currency in which the fee will be calculated.

  ⓘ The *Fee Curr* field is filled in with data from the *Settl Curr* field automatically if even one of the following fields is filled in: *Fee Base*, *Fee Min*, *Fee Max*. If the *Settl Curr* field is not filled in, an error message will be displayed.

- The *Fee Base* field specifies an additional mandatory fee and does not depend on the fee rate.

  ⚠ The system does not support processing negative values of the *Fee Base, Fee Min* and *Fee Max* fields. Positive values must be specified in these fields.

- The *Fee Min* field shows the minimum amount of the fee that will be charged regardless of the transaction amount.

- The *Fee Max* field indicates the maximum allowed amount of the fee that can be charged on a transaction.

- The *Fee %* field contains the fee rate. Note that the system can calculate the fee rate using one of two algorithms:

  ▪ The fee is calculated from the entire transaction amount. In this case, a positive value is set in the field.

  If the field value is positive, the fee is calculated according to the following equation:

$$Fee = min(Fee_{max}, \max(Fee_{min}, \frac{Amount_{settl} \cdot Fee_{\%}}{100} + Fee_{base})) \Big|_{Fee_{\%} \geq 0}$$

  For instance, when a transaction is performed, the client's account is debited for the transaction amount and the fee calculated according to this formula.

  ▪ The fee is calculated from the transaction amount including the additional fee. In this case, a negative value is set in the field.

  If the field value is negative, the fee is calculated according to the following equation:

$$Fee = min\left(Fee_{max}, \max(Fee_{min}, \frac{-Fee_{\%} \cdot (Amount_{settl} + Fee_{base} \cdot direction_{fee})}{100 - Fee_{\%}} + Fee_{base})\right) \Big|_{Fee_{\%} < 0}$$

  In this formula, the *direction $_{fee}$* coefficient depends on the value in the *Fee Dir* field. If the field value is "Debit", the coefficient is "-1"; if the value is "Credit", the coefficient is "1", and if "None", it is zero.

  An example of how a fee is calculated from the transaction amount including the fee amount:

  A bank branch, when servicing a card issued by another bank branch, debits the transaction amount and the transaction fee for that type of operation. The bank policy, however, specifies that the fee must be transferred to the account of the bank's head office. Since the head office receives information indicating the transaction amount including the fee, it is necessary to calculate the fee itself. A negative value in the *Fee %* field allows this process to be carried out automatically.

- The *FX Rate Type* is a mandatory field that shows the foreign exchange rate. The values "Middle" and "Buy/Sell" are possible.

- The *FX Type* field shows the type of foreign exchange (see the section "FX Types" in the Currency Conversion Administrator Manual).

- The *Increase %* field shows the percentage by which authorization and clearing amounts sent to the issuer are increase; this parameter is usually used to surcharge the transaction amount by an amount which is charged to the cardholder and counted as revenue for the acquiring bank.

ⓘ The field is only analyzed in the Source Service (the acquirer analyzes the field when posting a transaction on the acquirer's side).

- The *Fee Tariff* field contains a drop-down list of registered tariff types with the "Service" role.

- The *Limit Tariff* field contains a drop-down list of registered tariff types with the "Service Limit" role.

- The *VD Tariff* field contains a drop-down list of registered tariffs with the "Service Value Days" role.

ⓘ If the bank's distribution package does not include the WAY4 Tariffs module, the *Fee Tariff, Limit Tariff,* and *VD Tariff* fields are empty. The WAY4 Tariffs module is not included in the basic WAY4 configuration and is supplied under a separate agreement with the WAY4 system vendor.

Posting

This group of fields is used to create document posting rules:

- The *Fee Code* field is a code set for Nostro contract Services used to interact with payment systems (Visa, MasterCard, etc.). The value of this field is determined by payment system rules. This code defines what algorithm is used by payment systems to calculate fees. The value is selected from a list (handbook). The handbook of codes is generated on the basis of codes already entered in WAY4 using the menu item "Full → Configuration Setup → Products → Generate Service Fee Codes".

- The *Floor Limit* field is a numeric value which determines the maximum allowed amount of a transaction that can be made without authorization.

- The *Value Days* field can be used in two ways:

  ▪ If the *Value Days* field is filled in for a Service (main Service or for a Custom Fee), this value will determine the due normalisation period for the corresponding account (the period can be set using a tariff with the "Service Value Days" role). A non-empty value in the *Value Days* field (or a non-empty value of the tariff with the "Service Value Days" role) has a higher priority than the *Due Period* field in the account template (or the tariff with the "Ageing" role set in the template). However, specifics for calculating the period are determined in the account template (for example, specifying the normalisation date as a fixed day of the month, quarter or billing cycle).

    For example, for Source Services used to post transactions to accounts with the "Value Days Due" due normalization type, this field specifies the number of banking (business) days after which the acquirer reimburses the merchant. I.e. WAY4 uses a business calendar to differentiate between business days and weekends/holidays (see the "Business Calendar" section in the WAY4™ Dictionaries Administrator Manual).

  ▪ For Miscellaneous Services, this field determines the time shift, in days, for recurring payments (fees).

    ⓘ Note that if a Misc Service is used to post a transaction to an account with a certain due normalization type (*Due Type* field of the account template), the

Value Days field will affect calculation of the account normalization date. In this case, configure a custom fee for the main Misc Service. Note that:

♦ The main Misc service determines the date for charging the fee (by filling in the *Value Days* and *Charge Event* fields, see the description below).

♦ Fee parameters (amount, account to which the fee will be transferred, etc.) are configured in the custom fee. In doing so:

- The *Value Days* field is not filled in for the custom fee.

- To avoid inheriting the *Value Days* field value from the main Service, the OWN_VALUE_DAYS; tag should be set in the custom fee.

    In this case, account normalization will be performed according to rules set in the account template.

▪ The type of shift depends on the value of the "Charge Event" parameter of the fee type (see "Full → Configuration Setup → Transaction Types → Fee Types", the *Charge Event* field). The *Value Days* field is used with the following values of the "Charge Event" parameter:

♦ "Weekly". If the frequency of the Event when a fee is charged is specified in the "Fee Types" form (Full → Configuration Setup → Transaction Types → Fee Types) as "Weekly" the value of the *Value Days* field is shown as a number between 1 and 7, where 1 is Monday.

♦ "Monthly". If the value of the "Charge Event" parameter is "Monthly", the value in the *Value Days* field can be either positive or negative.

    Fee day ≥ 0 – the shift is calculated ahead for the given number of days from the first day of the current month.

    Fee day < 0 – the shift is calculated back for the given number of days from the first day of the next month.

♦ "Billing Date". If the value of the "Charge Event" parameter is "Billing Date" the value in the *Value Days* field can be either positive or negative.

    Fee day ≥ 0 – the shift is calculated ahead for the given number of days from the first day of the current billing cycle.

    Fee day < 0 – the shift is calculated back for the given number of days from the first day of the next billing cycle.

♦ "Yearly". When the value of the "Charge Event" parameter is "Yearly" the value of the *Value Days* field determines the shift in months from the date the contract was opened and can be from "-12" to "12" inclusively. For example:

Fee day = "0". The shift of the first payment is one calendar year from the opening date of the contract. Subsequent payments are made annually on that date. If the date falls on a non-working day, payment is made on the first banking day following this date.

Fee day = "-12". The first payment is made, without a shift, on the date the contract is opened. Subsequent payments are made annually on that date. If the payment date falls on a non-working day, payment is made on the first banking day following this date.

Fee day = "6". The shift of the first payment is 1.5 calendar years from the opening date of the contract. Subsequent payments are made annually on that date. If the payment date falls on a non-working day, payment is made on the first banking day following this date.

- Service *Details* – this field is used for additional configuration of Services using tags (see the section "Tags Used when Configuring Service Packages and Services").

  ⓘ If two Services in a Package differ only by the value of the *Service Details* field (for example, in one Service the field is filled in with a tagged parameter and in the other Service the field is not filled in), these Services are considered identical and the Service Package will not be approved. These Services should be given different priorities.
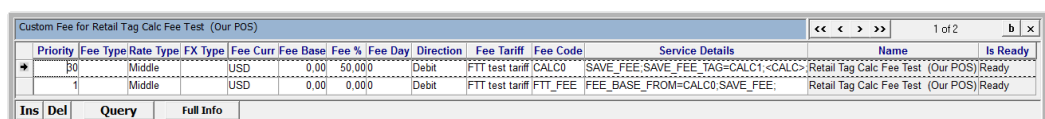
- The *Account Type* field specifies the account of the contract on which transactions will be made and to which fees will be charged.

- The *Account Curr* field shows the currency of the account specified in the *Account Type* field.

- The *Contract Type* field shows the contract type.

- *The Fee Contract* field shows the contract whose accounts will be used for transferring fees.

- The *Fee Account* field shows the account of the contract specified in the *Fee Contract* field which will be used to pay fees.

Changes to Services are approved by clicking the [Approve] button in the "Service Packs" form.

Custom Fee

Sometimes a situation arises when in addition to the specified fee, an additional fee is also charged, which is calculated as a percentage of the amount of an operation without taking into account the main fee. An example of such a situation is when a cardholder withdraws cash in a currency different from the account currency. In this case, the cardholder's account is debited for the transaction amount (amount of cash withdrawn), the transaction fee, plus a foreign exchange fee.

The system allows this type of fee to be processed automatically. To do so, the fee must be described in the "Custom Fee" form (see Fig. 9), opened by clicking the button of the same name in the forms "Source" (see Fig. 5 in the section "Source Services") and "Target" (see Fig. 6 in the section "Target Services").

Priority	Fee Type	Rate Type	FX Type	Fee Curr	Fee Base	Fee %	Fee Day	Direction	Fee Tariff	Fee Code	Service Details	Name	Is Ready
30		Middle		USD	0,00	50,000		Debit	FTT test tariff	CALC0	SAVE_FEE;SAVE_FEE_TAG=CALC1<CALC>	Retail Tag Calc Fee Test (Our POS)	Ready
1		Middle		USD	0,00	0,000		Debit	FTT test tariff	FTT_FEE	FEE_BASE_FROM=CALC0;SAVE_FEE;	Retail Tag Calc Fee Test (Our POS)	Ready

Fig. 9. Form for entering and editing custom fee information

This form contains the following fields:

- *Priority* – determines the procedure for charging custom fees set up in the "Custom Fee for…" form. The priority value is a number (0,1, etc.), and the larger the value, the higher the priority. I.e. fees are processed in descending

order of priority. Additional fees with a zero priority are processed last in random order.

- *Fee Type* – selection of the fee type from a drop-down list of non-transaction fee types available for the contract category specified for the given Service Package (the list is set in the "Full → Configuration Setup → Transaction Types → Fee Types" form).

- *Rate Type* – the foreign exchange rate. This is a mandatory field, whose value can be "Middle" or "Buy/Sell".

- *Fee Curr* – the fee currency.

- *Fee Base* – an additional mandatory fee which does not depend on the fee rate.

- *Fee %* - the fee rate.

- *Fee Day* – this field make it possible to determine *Value Days* for an additional fee regardless of settings for the main fee. The value of the *Fee Day* field fully corresponds to the value of the *Value Days* field. To use this setting, it is necessary to specify the tag OWN_VALUE_DAYS in the *Service Details* field of the custom fee (for more details see the section "Tags Used when Configuring Service Packages and Services").

- *Direction* – a mandatory field that determines the direction of the fee, where the value "Debit" indicates that the client is charged a fee for the operation, the "Credit" value specifies that the client is paid a fee for an operation, and "None", where no fee is charged.

- *Fee Tariff* – selection from a drop-down list of registered tariff types with the "Service" role. If the bank's distribution package does not include the WAY4 Tariffs module, the field is empty.

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

- *Fee Code* – the fee code. This field is used for interaction with external systems and when it is necessary to reflect the given fee as a separate line in a statement of accounts. In the latter case, in addition to filling in this field, it is necessary to set the parameter "P_SHOW_CUSTOM_FEE" of the corresponding report (see the section "Contract Statement Parameters" in the Cardholder Statements User Manual). The *Fee Code* field value is selected from a list (handbook). The handbook of codes is generated on the basis of codes already entered in WAY4 using the menu item "Full → Configuration Setup → Products → Generate Service Fee Codes".

- *Is Ready* – the *Is Ready* field shows that changes made to the given Service were approved.

Click the [Full Info] button to access the form with full information about additional fees (see Fig. 8 in the section "Full Information about a Service" for a similar form).

Click the [Approve] button in the "Service Packs" form to approve changed Services.

Chapter 3. Working with Service Packages

Working with Service Packages includes the following:

- Current Service Parameters (Approved Services)

- Copying Service Packages

- Configuring Additional Service Packages

- Configuring Usage Limiter Templates

- Configuring Events

- Configuring Message Templates (Group Msg)

- Configuring Preferred Counterparties (Preferred)

- Entering and Editing Tags

- Checking Service Packages

- Approving Service Packages

Current Service Parameters (Approved Services)

The forms "Target for ..." (see Fig. 6 in the section "Target Services"), "Source for ...." (see Fig. 5 in the section "Source Services") and "Miscellaneous for ..." (see Fig. 7 in the section "Miscellaneous Services") show the values of Service parameters saved after the last configuration. These values are not necessarily the current ones, as it is possible to set parameter changes for a future date (see the section "Approving Service Packages").

Click the [Approved] button in the form with additional information about a Service Package (see the section "Additional Parameters of Service Packages") to open the special form "Approved for <name of Service Package>" (see Fig. 10). This form contains a list of the current values of Service parameters.

This operation is not available for additional Service Packages and child Service Packages (linked to a parent Service Package through the *Parent Pack* field). When it is necessary to view current parameter values of a child Package, use the "Approved for…" form of the parent Service Package.

Fig. 10. Form for viewing the list of active Services in the Service Package

The "Approved for <name of Service Package>" form contains the following fields:

- *Service Class* – the transaction class according to the bank classification, for example:
  - The value "Transaction" means that the Service is a card transaction.
  - The "Misc" value means that the Service describes fee charging/payment.
- *Src/Tgt* – this field can have one of two values:
  - "Source" means this Service is used for operations in which the contract used by the given Service Package is the source of transaction information.
  - "Target" means this Service is used for operations in which the contract used by the given Service Package is the target of transaction information.
- *Service* – Service name.
- *Allowed* – this field can have one of the following values:
  - "Always" – the Service permits transactions to be executed.
  - "Offline on Issuing" – the Service prohibits authorizations, but processes financial documents registered in the system without authorization according to Service rules using a dispute contract.
  - "Offline on Acquiring" – the Service permits acquiring operations offline only (this Service is like the "Credit" operation in the MasterCard payment system).
  - "Never" – the Service prohibits authorization and processing of financial documents.
- *Fee Dir* – a mandatory field that determines the direction of a fee, where the value "Debit" means that the client is charged a fee for the transaction; "Credit", the transaction fee is paid to the client, and "None", no fee is charged.
- *Fee Curr* – fee currency.
- *Fee %* - fee rate.
- *Base* – an additional mandatory fee that is charged and does not depend on the fee rate.
- *Min* – the minimum amount of the fee charged regardless of the transaction amount.
- *Rate Type* – the type of foreign exchange rate.
- *Account Type* – the type of contract account on which transactions are made and fees are charged.

Clicking the [Full Info] button opens the form containing full information about a Service in viewing mode (see Fig. 8 in the section "Full Information about a Service" for a similar form). The [FI Parms] button of this form is used to show changes to Service Package parameters that are made when the given Service Package is transferred to another financial institution.

Current Service parameters are accessible from the form with full information about a Service (see the section "Full Information about a Service") by clicking the [History] button to open a form with full information about a Service in viewing mode.

Copying Service Packages

WAY4 allows users to copy Service Packages. This function ensures that users do not have to spend a lot of time setting up parameters for new Service Packages based on existing ones.

Service Packages are copied:

- In the form with the Service Package's main parameters (see Fig. 2 in the section ""Service Packs" Form") by clicking the [Approve] button and selecting "Duplicate" from the context menu.

- In the form with additional information about the Service Package (see Fig. 3 in the section "Additional Parameters of Service Packages"). Click the [Actions...] button and select the value "Duplicate" from the context menu.
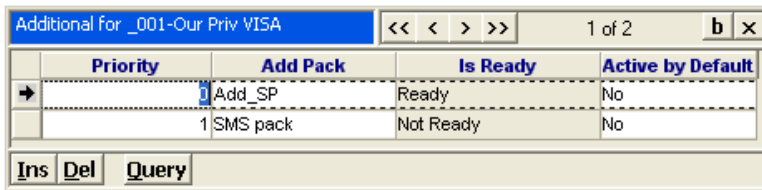
After copying, a new Service Package will be added to the list of Service Packages. It will differ from the original by an underscore added as a prefix to the Service Package name.

ⓘ WAY4 allow allows copying of separate Services within a Service Package. This is possible as follows:

- In the form with the main information about Services (see the section "Service Types") using the [Duplicate] button.

- In the form with full information about a Service (see Fig. 8 in the section "Full Information about a Service"). Click the [Actions...] button and select the "Duplicate" value from the context menu. The Service parameters are copied in full.

Configuring Additional Service Packages

WAY4 allows new Service Packages to be created as a combination of existing Service Packages. This considerably decreases the number of management points for tariffs and limiters. To create an additional Service Package, click the [Additional] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form"). This displays the "Additional for <name of Service Package>" form (see Fig. 11)

Fig. 11. Form for entering information on an additional Service Package

This form contains a list of additional Service Packages for the selected Service Package (package marked as "Additional", see the section "For Contracts") and may contain Risk Packages (package marked as "Risk Rule", see the section "For Contracts").

ⓘ Service Packages for risk control are created and activated in special forms of the monitoring module (see the document "Risk Monitoring"). The "Additional

for…" form can be used to view additional Service Packages for risk control, it is not recommended to edit settings for their activation in this form.

To add a new additional Service Package (with the "Additional" marker) in the form "Additional for ...", click the [Ins] button and fill in the following fields of the new record:

- *Priority* – Service Package priority level used to resolve conflicts between Services of different additional Service Packages describing the same transaction; the priority value is specified in numeric form (0, 1, etc.) with a larger value being of higher priority: in the event of a conflict, the system chooses the Service Package with the highest priority.

- *Add Pack* – the name of the additional Service Package.

- *Active by Default* – this field has two possible values:

    - "Yes" – the additional Service Package will be active by default when the corresponding main Service Package is selected during contract creation or editing.

    - "No" – the additional Service Package will not be used when the corresponding main Service Package is selected; activation of this additional Service Package is possible through the configuration of Events (see the section "Activating Additional Service Packages" in the Events Administrator Manual).

Note that only Service Packages with the value "Additional" in the *For Contracts* field of the "Service Packs" form and a contract type (*Contract Type* field) which corresponds to the main Package can be used as an additional Service Package. However, the value of the *Client Category* field for the main and additional Service Package may differ.

Moreover, Service Packages with the "Additional" value can only be connected to main Service Packages (those that have the "Product" value in the *For Contracts* field of the "Service Packs" form).

The list of additional Service Packages for a certain contract is accessible in forms for viewing and editing contract data (see the Customer Service User Manual).

If a usage limiter template (see the section "Entering and Configuring Limiter Templates" in the Usage Limiters Administrator Manual) is configured for an additional Service Package, and the code of this template does not correspond to any codes of limiter templates in the main Service Package, this template will be inherited by the main Service Package. If the codes coincide, both limiters will be activated.

After filling in the form, click the [Approve] button in the "Service Packs" form to approve changes to the Service Package.

In the *Special Parms* field, conditions can be configured for attaching Additional Service Packs according to set client and contract classifier values (using the IF_CS group tags). For more information, see the section "Executing Actions Depending on Classifier Values" in the document "WAY4™ Client and Contract Classifiers").

When attaching a Service Package, preferred counterparties set up for it are inherited to the main Service Package. If an attached Service Package is deleted from the list of attached Service Packages (from the "Additional for <Service Package> name" form (see Fig. 11)), inherited preferred counterparties are deleted from the main Package.

Until version 03.41.30, inherited preferred counterparties were not deleted from the main Package when an attached Service Package was deleted. For configurations set up before version 03.41.30, use the menu item "Full → Configuration Setup → Products → Purge Utilities → Check service pack preferred" to check preferred counterparty settings in Service Packages. If the check finds Service Packages with "invalid" inherited preferred counterparties, the corresponding message will be shown (these Service Packages are listed in the process log. Preferred counterparties inherited from a deleted attached Package will be deleted from the main Service Package when changes to the main Service Package are approved.

Configuring Usage Limiter Templates

WAY4 allows various limits on transaction activity to be used for contracts. Such limits are set using contract usage limiters.

In Service Packages, limiter parameters are set in general templates. These templates determine limiter parameters for all contracts using the corresponding Service Package.

To configure a limiter template, click the [Usage] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form"). For more information about configuring limiter templates, see the section "Entering and Configuring Limiter Templates" in the Usage Limiters Administrator Manual.

Configuring Events

The system allows the activation or deactivation of the link between an additional Service Package and the Service Package of contracts when an Event occurs.

To set such an Event, use the "Events for <name of Service Package>" form opened by clicking the [Events] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form"). For more information on configuring this type of Event, see the section "Activating Additional Service Packages" in the Events Administrator Manual.

Configuring Message Templates (Group Msg)

WAY4 allows information or marketing messages to be sent to a client whose contract uses the given Service Package.

To configure message templates, click the [Group Msg] button in the "Service Packs" form (see Fig. 2 in the section ""Service Packs" Form".) For more information about message templates, see the "Configuration of Client Messages" document.

Notification that a Misc Fee will be charged

This setting makes it possible to open an Event for a specified number of days before a Misc fee is to be charged and to send the corresponding message to the client.

Setup is performed with the EVENT_BEFORE=<Event type> and <EVENT_BEFORE_DAYS=<number of days> tags in the Misc service or transaction subtype.

ℹ Note that for annual fees, the EVENT_BEFORE_DAYS tag can be specified only in a transaction subtype.

One Event type can be used for notification about any Misc fees, since information about the upcoming fee goes into the Event's tags when opening it (in the Event log – the USAGE_ACTION table).

Events for notification are processed in the standard way when the banking date opens. If there are weekends/holidays, notifications about fees for several days can be sent using one Event.

Example 1. The value of the EVENT_BEFORE_DAYS parameter is "3'. A standard calendar is used, in which Saturday and Sunday are holidays. When Monday is opened, fees can be charged that fall on Saturday, Sunday, and Monday. Events to notify about the fees for all these days are opened as though for Saturday – when Wednesday is opened.

Example 2.

- The value of the EVENT_BEFORE_DAYS parameter is "3".

- There will be holidays from 31/12/2016 to 08/01/2017. The business date 09/01/2017 is opened in the evening of 30/12/2017. All fees will be charged that fall in the period from 31/12/2016 to 09/01/2017.

- Notifications for all these fees will be sent on 28 December. Notifications will not be sent on 29 December and 30 December (since they were already sent 3 days before the holidays started).

- When 09/01/2017 is opened, notifications are sent for fees to be charged from 10 to 12 January (since 10 and 11 are weekends and notifications could not be sent earlier, since banking days were not opened).

ℹ When the procedure for opening the day is performed at the start of the business day, it is recommended to use a longer period for notification (for example, 5 days instead of the 3 days in our example). This makes it possible to avoid delays in sending notifications when there are weekends/holidays. For example, in our example, when Monday is opened on Monday morning, notifications about fees to be charged on Tuesday will only be sent on Monday.

Configuring Preferred Counterparties (Preferred)

To expand the functionality of Services, preferred counterparties (Preferences) are used, which can be defined on the Service Package level. To configure the list of counterparties, click the [Preferred] button in the "Service Packs" form (see Fig. 2

in the section ""Service Packs" Form"). For more information about preferred counterparties, see the Preferred Counterparties Administrator Manual.

Entering and Editing Tags

The "Details for <name of Service Package> and "Full Info for <name of Service>" forms contain special fields for entering and editing tags. These are the *Special Parms* and *Service Details* fields, respectively.

To optimise the process of entering and editing tags in the given fields, the "Tagged Data" form (see Fig. 12) is used. This form is opened by clicking the [Tagged Data] button in the "Details for <Service Package name>" and "Full Info for <Service name>" forms.

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

Fig. 12. The "Tagged Data" form

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

- The name of the tag in the *Tag* field can be selected from a system list of tags. If the tag is absent from this list, its name can be entered manually.

- The value of the tag is entered in the *Value Data* field.

- The *Value Tag* field:

  ▪ When entering tag parameters in the "Tagged Data" form, it is necessary to select the value "Tag Present" in the *Value Tag* field for tag parameters to be saved correctly. After saving, tag parameters will be displayed in the corresponding field of the higher-ranking form (in the *Special Parms* or *Template Details* field).

  ▪ When the "Tag Absent" value is selected in this field, the record of this tag will be deleted from the "Tagged Data" form, as well as from the corresponding field of the higher-ranking form, after changes are saved.

- The *Value Type* field is used to specify the type of tag value entered in the *Value Data* field:

  ▪ "CheckBox" – in this case, it is not necessary to fill in the *Value Data* field.

  ▪ "String" – a string value.

  ▪ "Counter" – the tag value is entered as a whole number (from "0" to "9").

  ▪ "Tag" – the tag value is entered as one symbol (usually "Y" or "N").

  ▪ "Money" – a numeric value.

  ▪ "Currency" – the tag value is entered as a numeric currency code.

  ▪ "Unknown" –the tag value is not set although it is mandatory to enter a value in the *Value Data* field.

  ▪ "List" – the tag value can be set as a list of values. In initial manual setup, a comma-delimited list of values is specified in the *Value Data* field (or in the

Special Parms and *Service Details* fields 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 displays the description of the tag whose name is selected from the system list.

- The *Is Ready* field displays the results of checking tag parameters:
  - The value "Ready" is displayed if the check was successful.
  - The value "Not Ready" is displayed if errors were found during the check.

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

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

Tag parameters entered in "Tagged Data" can be saved by clicking the [Actions...] button and selecting an item from the context menu:

- "Save Tagged Data" (in the "Details for <name of Service Package>" form).

- "Save Tags" (in the "Full Info for <name of Service>" form).

Checking Service Packages

Service Package parameters are checked in the screen form with additional information about a Package (see Fig. 3 in the section "Additional Parameters of Service Packages") by clicking the [Actions...] button and selecting the value "Check" from the context menu.

If errors occur when checking Service Package parameters, a window with the "Error" heading will appear on the screen. Information about errors can be found in the process log by selecting "Full → Process Log → Last Process" from the user menu. This menu item is used to open the "Last Process" form containing information about the last system process.

Approving Service Packages

After creating or changing Service Package parameter values, click the [Approve] button in the Service Package grid forms (see Fig. 2 in the section ""Service Packs" Form" or Fig. 4 in the section "Special Forms for Working with Service Packages") to approve changes. Note that it is possible to approve changes to a Service Package from a date different from the current date. To do so, specify the corresponding date in the form which appears when the [Approve] button is clicked (see Fig. 13).
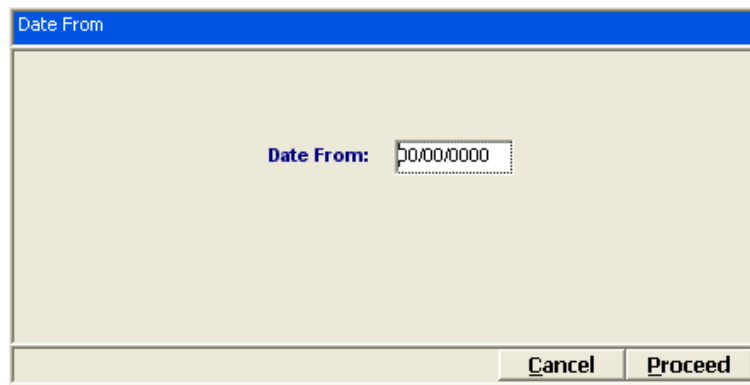
Fig. 13. Form for setting the date when approving a Service Package.

Approved changes to a main Service Package will be active beginning from the date indicated. If this date is before or on the current banking date, changes will be active immediately. If this date is after the current banking date, the approved changes will come into effect only after the opening of the corresponding banking day.

⚠ Note that this date affects activation of the main Package's Services. Settings for Events and Additional Service Packages become effective immediately after approval of the main Package, regardless of this date.

ℹ A check is performed when approving a Service Package. If two Services in a Package differ only by the value of the *Service Details* field (for example, in one Service, the field is filled in with a tagged parameter and in the other Service, it is not filled in), the Services are considered identical and the Service Package will not be approved. These Services should be given different priorities (see the *Priority* field).

⚠ Note that if changes to a Service Package's Service will become active in the future, the current value of this Service's parameters are available in the list of active Services (see the section "Current Service Parameters (Approved Services)").

Chapter 4. Tags Used when Configuring Service Packages and Services

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 *Special Parms* field of a Service Package		
ACTIVITY_DELAY	<number of days>	This parameter allows the following mechanism to be implemented: if the date of the contract's last financial activity (this information is taken from the ACTIVITY_DATE tag automatically inserted in the contract's ext_data field) is less than the current banking date for <number of days>, an Event with the code CONTRACT_DELAY is opened for the contract. The ACTIVITY_DATE tag is filled in when the ACTIVITY_FOR tag is set in the Service or subtype of the transaction. If the ACTIVITY_FOR tag is not specified, the date of the last activity on the contract is considered the contract opening date.
USE_AUTH_FX	"Y" "N"	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
STMT_FROM_SEE_MAIN		If this tag is specified, a mini-statement is not generated for this contract, but for the main contract in the "See Main" hierarchy. In the mini-statement, transactions are shown that were made for the main contract in the "See Main" hierarchy and for all its subordinate contracts.
BAL_TYPE_1, …BAL_TYPE_6	<balance type code>	These tags are used to specify the set of balance types about which information is sent in the response to a transaction request. These tags make it possible to redefine the global parameters BALANCE_TYPE_1, BALANCE_TYPE_2 (the default values are 02 (funds available) and 91 (credit limit)) and/or add the code of a balance type that is included in a mini-statement. For example, BALANCE_TYPE_3=91;. Balance type codes registered in WAY4 can be specified as values. A balance code must be a two-digit number. These tags can be set in additional Service Packages. When generating data on the state of contract balances that is sent in the response to an authorisation request, activation and priority of additional Packages are taken into consideration. Tags can be used together with the tags BAL_FX_TYPE, BAL_FX_RATE_TYPE. For international cards (that can be serviced in VISA, Mastercard networks), it is not recommended to redefine the BAL_TYPE_1 balance type (by default, the value of this balance type is "02"), since a redefined value that does not correspond to specifications is sent to the IPS in the response to an

Name	Value	Description
		authorization request and may not be recognized in the IPS. This limitation does not apply to local cards.
U_PBT;		This parameter allows automatic unblocking of new plastic on the first PIN transaction.
BAL_FX_TYPE	<FX type code>	In a balance inquiry, makes it possible to specify the balance's FX Type (Full → Configuration Setup → Accounting Setup → FX Types). Can be used together with the BAL_FX_RATE and BAL_CURR tags.
BAL_FX_RATE_TYPE	"B" "M"	Makes it possible to specify the FX rate type of a shown balance (in a balance inquiry): • "B" – Buy/Sell • "M" – Middle (default value) Used together with the tags BAL_FX_TYPE and BAL_CURR.
BAL_CURR	<Currency code>	In a balance inquiry, allows the currency of the balance to be set, regardless of the contract currency. Conversion will be made with consideration of the parameters BAL_FX_RATE_TYPE and BAL_FX_TYPE. Used together with the tags BAL_FX_TYPE and BAL_FX_RATE_TYPE.
USE_TRANS_AMOUNT	"Y" "N" "F"	Used if the global parameter of the same name or a tag on the financial institution level must be redefined for a certain Service Package of a financial institution (this parameter is used to specify the way an amount will be calculated that is debited from accounts of "our" cards when posting a financial document loaded from payment systems, as well as the amount of funds blocked when processing an authorization request): • "F" value – the transaction amount is blocked and debited if the card has an account in the transaction currency. • "Y" value – the transaction amount is always blocked and debited. • If the tag is not set or set to "N", the settlement amount will be debited from the card. For more information, see the description of the USE_TRANS_AMOUNT tag in the section "Additional FI Parameters" of the document "Financial Institutions". The tag can be set for Additional Service Packages.
FROM_AUTH	"SERVICE" "SERVICE_TGT" "SERVICE_SRC" "FEE_AMOUNT" "FX_RATE"	This tag specifies that if authorization was made on a Service from this package, the financial document for this transaction will inherit the parameters set by the tag from the Service on which authorization was made. Tag values are: • "SERVICE" – all Service parameters are inherited, meaning no Service will

Name	Value	Description
		be searched for when posting the financial document. When this value is set, both target contract Service parameters and source contract Service parameters are inherited. • "SERVICE_TGT" – target contract Service parameters are inherited. • "SERVICE_SRC" – source contract Service parameters are inherited. • "FEE_AMOUNT" – only the fee amount is inherited. • "FX_RATE" – FX rates at the time of authorization are inherited (regardless of the global parameter USE_AUTH_FX) All values can be set in the format: FROM_AUTH=SERVICE,FEE_AMOUNT,FX_RATE; The tag can be set on the transaction subtype level, but the tag in a Service Package will have a higher priority.
PARTIAL_APPROVAL_ENABLE	"Y" "N"	Redefines the global parameter of the same name. Allows partial authorizations to be prohibited. To do so, set the parameter value to ""N". In standard system behaviour (if the parameter is not set or set to "Y") if the request contains a flag that the merchant is prepared to accept partial authorisation (PARTIAL_APPROVAL_SUPPORT=Y;) and the amount available to the contract is less that that received, we correct the amount.
NO_SATELLITE_BALANCE		This tag is set when it is not necessary to send balance information in transaction request responses.
AUTH_USE_DOMESTIC	"Y" "C" "A" "N"	Tag can be used to redefine the value of the global parameter of the same name (AUTH_USE_DOMESTIC) and the tag of the same name on the financial institution level. The tag can be set for an Additional Service Package (for more details, see the document "WAY4™ Global Parameters").
IF_CS_TYPE	<Classifier type code>	This tag sets the type of classifier checked. Used together with the IF_CS_VALUE, IF_NOT_CS_VALUE, IF_CS_TYPE_FOR tags. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers". It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
IF_CS_VALUE	<CS_STATUS_VALUE.CODE 1>, <CS_STATUS_VALUE.CODE 2>,…,<CS_STATUS_VALUE.CODE N>	A payment order is only activated if clients and their contracts correspond to the set value of this classifier. Several codes separated by commas may be specified as the value of this tag. Used together with the IF_CS_TYPE tag. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers".

Name	Value	Description
		It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
IF_NOT_CS_VALUE	<CS_STATUS_VALUE.CODE 1>, <CS_STATUS_VALUE.CODE 2>,…,<CS_STATUS_VALUE.CODE N>	A payment order is not activated if clients and their contracts correspond to the set value of this classifier. Several codes separated by commas may be specified as the value of this tag. Used together with the IF_CS_TYPE tag. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers". It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
DAYS_TO_RENEW	<number of working banking days>	DAYS_TO_RENEW=<number of working banking days>; makes it possible to open an Event with the code CARD_AUTO_RENEW for a specified number of days until the card's expiry date. This Event can be used, for example, to configure sending notification to a client about scheduled card reissue.
NO_COPY		When this tag is specified, the link is broken between this Service Package and the Service Package specified in the Parent Pack field, i.e. after setting this tag and confirming the changes, the parameters of the "parent" Package will not be copied to this Package.
CHCK_BASE_USAGES	"Y" "N"	The tag CHCK_BASE_USAGES=<value>; redefines the global parameter of the same name: • If the value is "Y" (default value), when processing a request for a related card, the limiters of this card and of the base card are checked. • If the value is "N", earlier system behaviour is restored when only the related card's limiters are checked.
LOAD_IMMEDIATELY;		Used for special attached Service Packages for risk management ("Additional" or "Risk" package) requiring immediate loading to all "Product" Service Packages. In Approve of a Package with this flag, a check will be made for the absence of Services in it. If there are none, and there are no other errors, limiters from this Package will be imported to all Product packages to which this "Additional" or "Risk" Package is attached.
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.

Name	Value	Description
		The Reversal Management module is not included in the WAY4 basic configuration and is supplied according to a separate agreement with the WAY4 vendor.
IF_PARM	\<tagged parameter name\> CLIENT.SHORT_NAME CLIENT.FIRST_NAME CLIENT.LAST_NAME CLIENT.COMPANY_NAME CLIENT.AGE	This tag sets a checked classifier (tagged parameter) for a contract. Used together with the IF_PARM_VALUE tag. For more information, see the section "Classifiers without a Fixed List of Values" of the document "WAY4™ Client and Contract Classifiers". A number of client parameters can be checked. To do so, the following values can be specified as the tag's value: CLIENT.SHORT_NAME – check the Short Name field in the client form. CLIENT.FIRST_NAME – check the First Name field in the client form. CLIENT.LAST_NAME – check the Last Name field in the client form. CLIENT.COMPANY_NAME – check the Company Name field in the client form (place of work) CLIENT.AGE – client age. Determined according to date of birth (Date of Birth field) and the system date (i.e. age at the time of the request is determined). These client attributes do not have to be registered as contract custom parameters. It is possible to set a check of several parameters for a contract by using the IF_PARM_NUMB, IF_PARM\<N\>, IF_PARM_VALUE\<N\>, IF_PARM_FOR\<N\> tags. See the description of the IF_PARM_NUMB tag.
IF_PARM_VALUE	"\<tagged parameter value\>" "NOT_EMPTY" "EMPTY" "LIST_WITH:\<list of values separated by commas\>" "BETWEEN\<value\> AND \<value\>"	An Additional Service Package will be enabled if clients and their contracts correspond to the set classifier value. Used together with the IF_PARM tag. The IF_PARM_VALUE=NOT_EMPTY; tag is used to check for the existence of a parameter (the tag set using the IF_PARM tag) without checking its value. When the IF_PARM_VALUE=EMPTY; tag is set, the additional Package will be enabled if the specified parameter doesn't have a value. Note that with this value, the additional Package will also be enabled if the parameter is not present in the contract. When "IF_PARM_VALUE=LIST_WITH:\<list of values separated by commas\>" is set, the additional Package will be enabled if even one value from the list (LIST_WITH) matches that set in the contract. When "IF_PARM_VALUE=BETWEEN\<value\> AND \<value\>" the additional Package will be enabled if the value in the contract falls in the range set by the tag (for example, IF_PARM_VALUE=BETWEEN40 AND 50;). The value must be set with spaces, as shown in the example. It is possible to set a check of several parameters for a contract by using the

Name	Value	Description
		IF_PARM_NUMB, IF_PARM\<N>, IF_PARM_VALUE\<N>, IF_PARM_FOR\<N> tags. See the description of the IF_PARM_NUMB tag.
STMT_SERV_CLASSES	\<types of operation with Service Class classifier>	The tag sets operations (Service Class) included in a mini-statement. Service Class operation types are listed without delimiters. Possible values for the tag: • "T" (Transaction) – card transactions • "M" (Misc) – charge/pay fees • "I" (Interests) – interest accrual • "i" (Interest Fee) – charge interest fees • "U" (Upper Norm) – upper limit normalisation • "u" (Upper Limit Special) – special upper limit normalisation • "L" (Lower Norm) – lower limit normalisation • "l" (Lower Limit Special) – special lower limit normalisation • "D" (Due Norm) – due normalisation • "d" (Due Special) – special due normalisation • "A" (Account Transfer) – adjustment entries • "C" (Credit Limit) – operations setting and changing a credit limit • "R" (Rev/Exp) – operations related to daily interest accrual for an account or supplementary entries for accounts, for example, when paying loan interest. • "r" (Reserve for Bad Debts) – loan reserving • "c" (Additional Cr Limit) – operations setting and changing an additional authorisation limit and blocking the main one. • "P" (Online Payment) – operations related to ATM payments The tag redefines the global parameter of the same name
IF_CURRENCY	\<tag name> TRANS_CURR SETTL_CURR RECONS_CURR CONTRACT \<tag or contract custom parameter> \<classifier code> \<three-digit numeric currency code or currency name>	This tag sets the document currency that must be checked. Possible values: • Document tag containing the currency code. • TRANS_CURR – transaction currency from the document. • SETTL_CURR – settlement currency from the document. • RECONS_CURR – reconciliation currency from the document (in the currency in which the source bank provides transaction information to the payment system). • CONTRACT – contract currency. • Tag or contract custom parameter with a currency code. A contract's custom parameter can be registered as a "Primary" classifier. •Code of the classifier with a currency code. •Explicit specification of currency – the tag value may be a code or the name of a certain currency.

Name	Value	Description
		Used together with the IF_CURRENCY_VALUE, IF_CURRENCY_RULE, IF_CURRENCY_FOR tags.
IF_CURRENCY_VALUE	EMPTY NOT_EMPTY USE_IN_BANK <currency code1>, <currency code2>,…,<currency codeN> <currency name1>, <currency name2>,…,<currency nameN> <tag name>	An additional Service Package will be attached if the document currency specified with the IF_CURRENCY tag corresponds to parameters set in the IF_CURRENCY_VALUE tag. Possible values: • EMPTY – this currency is not specified. • NOT_EMPTY – this currency is specified. • USE_IN_BANK – this currency has the Use in Bank parameter value of "Yes" (i.e. for this currency, conversion can be performed in WAY4 and the rate of the currency to the financial institution's local currency can be entered during the daily opening procedure). • A code or name of a certain currency or several currencies, separated by commas can be used as the tag value. • Tag or contract custom parameter containing a currency code or list of currencies separated by commas can be used as the tag value. A contract custom parameter can be registered as a "Primary" classifier.
IF_CURRENCY_RULE	ACC_SCHEME EQUAL NOT_EQUAL NOT_IN_LIST	To set additional conditions for checking a currency, use the IF_CURRENCY_VALUE tag together with the IF_CURRENCY_RULE tag: • To check the currency set in the IF_CURRENCY for correspondence to the Accounting Scheme it is necessary to set the tags IF_CURRENCY_RULE=ACC_SCHEME;IF_CURRENCY_VALUE=<list of account codes separated by commas>;. If a list of account codes is set, the currency is searched for among the specified Accounting Scheme account templates. If a list of accounts is not set, the currency is searched for among all Accounting Scheme account templates. • To check a currency for correspondence (or non-correspondence) to another document currency, it is necessary to set the tags IF_CURRENCY_RULE=EQUAL; or IF_CURRENCY_RULE=NOT_EQUAL; the value of IF_CURRENCY_VALUE is also set as the value of the IF_CURRENCY tag. • When IF_CURRENCY_RULE=NOT_IN_LIST; is specified, a check will be made that the currency is not included in the list of currencies specified explicitly or through the parameter in the tag IF_CURRENCY_VALUE.
ADD_PACK_BT	"N"	The ADD_PACK_BT=N; tag makes it possible to not analyse additional Service Packages when generating balances returned in a response to an authorisation request. The tag is set in the main Service Package (Package used to create Products and register contracts). If this behaviour must be used

Name	Value	Description
		for all Service Packages, set the value of the global parameter ADD_PACK_BT to "N".
FROM_ORIG_MODE	"SERVICE" "FX_RATE"	Tag specifying that when posting a secondary document with a Service from this Package, the secondary document for this transaction will be posted with inheritance of parameters specified by the tag from the Service with which the original document was posted. The tag can have the following values: ▪ "SERVICE" – all Service parameters are inherited, meaning no Service will be searched for when posting the financial document. If a tariff was used when posting an authorization with this Service, a search will not be made for a tariff when posing the financial document - the tariff for authorisation will be used (i.e. parameters will be inherited from the corresponding tariff). ▪ "FX_RATE" - FX rates at the time of authorization are inherited (regardless of the global parameter USE_AUTH_FX) All values can be set in the format: FROM_ORIG_MODE=SERVICE,FX_RATE; The tag can be set on the transaction subtype level. If the tag is set in a subtype, it will not be searched for in a Package.
CS_DOMAIN	<list of domain codes separated by commas>	The tag makes it possible to specify classifiers available for a specific Service Package - i.e. these classifiers will be available for contracts with this Package. The tag value is the classifier domain code or list of domain codes separated by commas. A domain code mask using the "[%]" service character can be used as the tag value. The service character "_" meaning "any character" can be used in the tag value. Note the following: • Classifiers from the domain set in the Service Package are available for contracts created using this Package. I.e. for the main contract in a Main/Sub hierarchy, classifiers from the domain set for a subcontract (and vice versa) are not available. • Classifiers from domains set in Additional Service Packages are not available for contracts. • If a calculated classifier (Decisions) is based on classifiers, these classifiers must be available for the contract (i.e. must be included in the domain set for the financial institution or Package. The tag can be set for financial institutions.
CS_SKIP_LIST	<list of classifier codes or groups of codes separated by commas>	The tag value is a comma-separated list of classifier codes (code field) or group codes (group_code) that do not need to be shown. As the tag value, specify the classifier codes (code field) or groups of codes (group_code), separated by commas, that will be unavailable for a contract

Name	Value	Description
		with this Service Package. I.e. these classifiers will not be shown in the list of classifiers that can be set.
IF_PARM_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	The IF_PARM_FOR tag is used together with the IF_PARM/IF_PARM_VALUE tags to redefine the contract for which these checks are made. • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the document's *Source Contract* (source_contract) field (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract specified in the document's *Target Contract* (target_contract) field (see the "Doc-Brief" form). • "FROM_DOC" – the contract is taken from the document's *Add Data* (add_info) field, according to the tag specified with the CONTRACT_TAG tag (in the same field). • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_PARM_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the

Name	Value	Description
		LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the IF_PARM_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. It is possible to set a check of several parameters for a contract by using the IF_PARM_NUMB, IF_PARM\<N\>, IF_PARM_VALUE\<N\>, IF_PARM_FOR\<N\> tags. See the description of the IF_PARM_NUMB tag.
IF_CS_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=\<tag name\> "LIAB_CATEGORY"	The IF_CS_TYPE_FOR tag is used together with IF_CS group tags to redefine the contract for which these checks are made. • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – from a higher-ranking contract in a "Liability" hierarchy. • "TOP" – from the top contract in a hierarchy. • "BASE" – from the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the document's *Source Contract* (source_contract) field (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract specified in the document's *Target Contract* (target_contract) field (see the "Doc-Brief" form). • "FROM_DOC" – the contract is taken from the *Add Data* (add_info) field. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=\<role code\> tag in this field; for example: IF_CS_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=\<role code\>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected.

Name	Value	Description
		• When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the IF_CS_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. It is possible to set a check of several parameters for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N>. See the description of the IF_CS_NUMB tag.
IF_CURRENCY_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC"	Redefines the contract for which a check is made using the IF_CURRENCY tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the *Source Contract* field of the

Name	Value	Description
	"RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	document (source_contract) of the document (see the"Doc-Brief" form). • "DOC_TARGET" – for the contract in the *Target Contract* field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's *Add Data* field (add_info) according to the tag specified using the CONTRACT_TAG tag in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_CURRENCY_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code> tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the IF_CURRENCY_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy,

Name	Value	Description
		starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
IF_CS_NUMB	<number of additional checks>	Sets the number of additional checks for the IF_CS_TYPE tag. The tag is used together with the IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. I.e. by default the IF_CS_TYPE tag is checked. If the IF_CS_NUMB=<number of additional checks>; tag is set, the specified number of IF_CS_TYPE<N> tags is checked. For example, if IF_CS_NUMB=3;, the IF_CS_TYPE1, IF_CS_TYPE2, IF_CS_TYPE3 tags are checked in addition to the IF_CS_TYPE tag.
IF_PARM_NUMB	<number of additional checks>	Sets the number of additional checks for the IF_PARM tag. The tag is used together with the IF_PARM<N>, IF_PARM_VALUE<N>, IF_PARM_FOR<N> tags. I.e. by default the IF_PARM tag is checked. If the IF_PARM_NUMB=<number of additional checks>; tag is set, the specified number of IF_PARM<N> tags is checked. For example, when IF_PARM_NUMB=3;, the IF_PARM1, IF_PARM2, IF_PARM3 tags are checked in addition to the IF_PARM tag.
ANY_CONTR_TYPE		The ANY_CONTR_TYPE; tag is set in an additional Service Package with the ATTACHMENT tag. The additional Package becomes available for attaching to any main Package with the same contract category (Card, Account, Device), but with a different contract type. In this case, when searching for and attaching Services from this Package, values of the Contract Type and Transaction Type fields are dynamically substituted with the parameters of the main Package's corresponding Services. By default (without setting this tag), only Packages with "Additional" in the For Contracts field of the "Service Packs" form and a contract type (Contract Type field) that matches the main Package can be attached to a Service Package. The tag is used together with the ATTACHMENT tag when configuring template additional Services. To limit the list of contract types to which a Service can be attached, use the CONTR_TYPE=<value>; tag in the additional Service.
IF_DATE	"CLIENT.BIRTH_DATE" "CLIENT.DATE_EXPIRE"	IF_DATE group tags make it possible to configure actions (in this case, attaching an Additional Service Package) depending on contract/client dates.

Name	Value	Description
	"CLIENT.DATE_OPEN" "CLIENT.ADD_DATE_01" "CLIENT.ADD_DATE_02" "CONTRACT.FIRST_ACTIVITY _DATE" "CONTRACT.DATE_OPEN" "CONTRACT.LAST_BILLING" "CONTRACT.NEXT_BILLING" "CONTRACT.DATE_EXPIRE" "CONTRACT.CARD_EXPIRE"	Contract/client dates can be compared with other dates (contract, client dates, system date, current banking date). Periods between dates can be checked according to various conditions. The IF_DATE tag sets the code of the date being checked, or the code of the date being compared with another date. Date codes can be contract/client dates specified in the following contract table (CONTRACT) or client table (CLIENT) fields: • "CLIENT.BIRTH_DATE" – a check is made according to the BIRTH_DATE field of the client's record • "CLIENT.DATE_EXPIRE" – a check is made according to the DATE_EXPIRE field of the client's record • "CLIENT.DATE_OPEN" – a check is made according to the DATE_OPEN field of the client's record (date on which the client was registered in the DB) • "CLIENT.ADD_DATE_01" – a check is made according to the ADD_DATE_01 field of the client's record • "CLIENT.ADD_DATE_02" – a check is made according to the ADD_DATE_02 field of the client's record • "CONTRACT.FIRST_ACTIVITY_DATE" – a check is made according to the FIRST_ACTIVITY_DATE field of the contract's record • "CONTRACT.DATE_OPEN" – a check is made according to the DATE_OPEN field of the contract's record • "CONTRACT.LAST_BILLING" – a check is made according to the LAST_BILLING field of the contract's record • "CONTRACT.NEXT_BILLING"– a check is made according to the NEXT_BILLING field of the contract's record • "CONTRACT.DATE_EXPIRE" – a check is made according to the DATE_EXPIRE field of the contract's record • "CONTRACT.CARD_EXPIRE" – a check is made according to the CARD_EXPIRE field of the contract's record The tag is used together with the IF_DATE_VALUE and IF_DATE_FOR tags. A check of several dates for a client/contract can be set using the IF_DATE_NUMB, IF_DATE<N>, IF_DATE_VALUE<N> and IF_DATE_FOR<N> tags. See the description of the IF_DATE_NUMB tag.
IF_DATE_VALUE	"EMPTY" "NOT_EMPTY" <Rule><Base Date Code><Period Rule>	The IF_DATE_VALUE tag sets rules for checking the date specified with the IF_DATE tag. Tag values: • "EMPTY" – the condition is met if the field for the date set with the IF_DATE field is not filled in.

Name	Value	Description
		• "NOT_EMPTY" – the condition is met if the field for the date set with the IF_DATE field is filled in. The field value is not checked. • The tag value can be composite and set in the following format (components are listed without delimiters): IF_DATE_VALUE=<Rule><Base Date Code><Period Rule>;, where: ▪ <Rule> – comparison rule. Possible values: ♦ "<" – less than ♦ ">" – greater than ♦ "<=" – less than or equal to ♦ ">=" – greater than or equal to ♦ If a comparison rule (<Rule>) is not set, the date is checked for matching (for the "=" value). ▪ <Base Date Code> – code of the date with which the date from the IF_DATE tag is compared. Possible values: ♦ Values that are described for the IF_DATE tag can be specified (i.e. dates from the contract or client table). ♦ "LDATE" – current banking date of the contract's financial institution ♦ "DB_DATE" – system (calendar) data (the system date is used without time) ▪ <Period Rule> – period that should be added or subtracted from the <Base Date Code> date. Set in the format <sign> <number of units ><period unit of measurement>, where ♦ <sign> – "+" or "-" (sign for adding or subtracting the specified period) ♦ <number of units> – number of days/months/years (see below) ♦ <period unit of measurement> – "D" (days), "M" (months), "Y" (years). Only one value can be set (i.e. one type of period is set in the tag). ♦ If the period <Period Rule> is not set, the two specified dates are compared according to the condition <Rule>. The tag is used together with the IF_DATE and IF_DATE_FOR tags. A check of several dates for a client/contract can be set using the IF_DATE_NUMB, IF_DATE<N>, IF_DATE_VALUE<N>, and IF_DATE_FOR<N> tags. See the description of the IF_DATE_NUMB tag. Examples of checks: Check of whether a client is already 40 years old: IF_DATE=CLIENT.BIRTH_DATE;IF_DATE_VALUE=<=DB_DATE-40Y; Check that a card has not expired: IF_DATE=CONTRACT.CARDS_EXPIRE;IF_DATE_VALUE=>=DB_DATE; Check that a card expires after a month: IF_DATE=CONTRACT.CARD_EXPIRE;IF_DATE_VALUE=<=DB_DATE+1M;

Name	Value	Description
IF_DATE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=\<tag name\> "LIAB_CATEGORY"	The IF_DATE_FOR tag is used together with the IF_DATE/IF_DATE_VALUE tags to redefine the contract/client for which the check is being made. Note. If the IF_DATE tag is used to check a date from the client table, first a search is made for a contract according to the IF_DATE_FOR tag, and then this date is checked for the corresponding client. Possible values for the IF_DATE_FOR tag: • "BILLING" – the check is made for the account contract used for settlement. • "LIABILITY" – the check is made for the higher ranking contract in the Liability hierarchy. • "TOP" – the check is made for the top contract in the hierarchy. • "BASE" – the check is made for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – the check is made for the contract specified in the *Source Contract* (source_contract) field of the document (see the form "Doc-Brief") • "DOC_TARGET" – the check is made for the contract specified in the *Target Contract* (target_contract) field of the document (see the form "Doc-Brief") • "FROM_DOC" – the check is made for the contract from the *Add Data* (add_info) field of the document according to the tag specified with the CONTRACT_TAG tag (in the same field) • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=\<role code\> tag in this field; for example: IF_PARM_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL ;. In this case, a search will be made for a contract with the CONTRACT_ROLE=\<role code\>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=\<tag name\>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field

Name	Value	Description
		using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as th LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the IF_PARM_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. A check of several dates for a client/contract can be specified using the IF_DATE_NUMB, IF_DATE<N>, IF_DATE_VALUE<N>, and IF_DATE_FOR<N> tags. See the description of the IF_DATE_NUMB tag.
IF_DATE_NUMB	<number of additional checks>	The tag sets the number of additional checks for the IF_DATE tag. The tag is used together with the IF_DATE<N>, IF_DATE_VALUE<N>, and IF_DATE_FOR<N> tags. I.e. by default, the IF_DATE tag is checked. If the IF_DATE_NUMB=<number of additional checks>; tag is set, the specified number of IF_DATE<N> tags is checked. For example, when IF_DATE_NUMB=3; in addition to the IF_DATE tag, the IF_DATE1, IF_DATE2, and IF_DATE3 tags are checked.
Tags in the *Service Details* field of a Service		
FROM_TRANS_DATE	<N of Days>	This parameter is used when interest must be accrued from the authorisation date (transaction date), not from the document posting date. The difference in days between the transaction date and posting date is taken into consideration – if this difference is less than the parameter value, interest is accrued from the transaction date; if it is larger – from the posting date. Entries related to limit normalisation generated as the result of processing this operation will also be posted with the Posting Date. If a Custom Fee is set up and it must also be posted from the transaction date, the tag for this fee must be specified separately, otherwise, the fee will be posted and processed with the Posting Date. The tag is only used on the Target Service side.

Name	Value	Description
		The tag can be set in the transaction subtype. The parameter's default value is "10".
SERV_EVNT	<EventCode>	This tag specifies that an Event with this code will be opened when executing an operation on this Service (when posting a financial document). If a fee activated according to an Event, in the case of a reversal, the fee will be reversed only if the REVERSED_BY_DOC tag is present in the Event type. It is recommended to use this tag in the transaction subtype.
<CALC>;		This tag means that the value of an additional fee, in whose settings this tag is specified is used only for calculating another fee (no entry is made); this Service is used together with Services containing the tags FEE_BASE_FROM=<fee code>; (percentage of fee amount) and SUBTRACT_FEE=<fee code> (the result is subtracted from the main fee). For example, for dividing up a sales tax (7% of the transaction amount) between the merchant (20%) and the state (80%), we configure three additional fees for the transaction Service: • Fee of 7% of the transaction amount, tag <CALC>; FeeCode=XXX;. • Fee 20% for the merchant, tag FEE_BASE_FROM=XXX;. This fee is calculated from the amount obtained for the additional fee with the <CALC>; tag. • Fee 80% for the state, tag FEE_BASE_FROM=XXX;. This fee is also calculated from the amount obtained for the additional fee with the tag <CALC>;.
<FIN>;		This fee is only charged when posting a financial document (financial debiting).
<AUTH>;		This fee is considered (blocked) only in authorisation.
<ADVICE_ONLY>;		A custom fee with this tag will only be charged when posting a document with the "Advice" request category.
<REVERSAL_ONLY>;		A custom fee with this tag will only be charged when posting a document with the "Reversal" request category. When setting fees charged for a reversal, the Direction field of the "Custom Fee" form must contain the "Credit" value.
<INCR>;		This tag allows a custom fee's Fee %, Fee Base, Fee Min and Fee Max parameter values to be used to specify the amount of the custom fee increasing or decreasing (depending on the fee direction) the document's Settlement Amount (in addition to main Service's Increase % parameter). The direction of the custom fee is set on the transaction type level, when the value

Name	Value	Description
		is "Credit, it is decreased (i.e. for "ATM", "Cash" and "Retail" transaction types the issuer is sent an amount more than the actual amount of the purchase; for "Note Acceptance" and "CH Payment" an amount less than the transfer amount). The Transaction Amount remains unchanged. The tag is used to configure acquirer fees (in device Source Services). Example of decreasing a document's Settlement Amount. Configure two Custom Fees with the following parameters: Custom Fee1 Fee %=<percentage by which we decrease> Direction=Debit Fee Code=PROMO (for example) Service Details=<CALC>; Custom Fee2. Fee %=0 Direction=Debit Service Details=<INCR>;SUBTRACT_FEE=PROMO; When these settings have been made, the following fees (for example) will be specified in the document: Transaction Amount=200USD Settlement Amount=140USD
<DECR>;		This tag is used to receive an acquiring fee from the transaction amount. This tag allows a custom fee that is debited from the cardholder to be sent to the acquirer's revenue account without increasing the transaction amount (the fee is taken from the transaction amount). For example, when paying for mobile phone services through a payment terminal, the service provider is sent a transaction message with the transaction amount decreased by the amount of the acquirer (payment terminal owner) fee. In the document, Transaction Amount and Settlement Amount remain unchanged. The changed amount sent to the service provider is registered in the Add Data (add_info) field as the value of the REST_SRC_AMOUNT tag. If the Settlement Amount must be decreased, use the <INCR> tag.
ATTACHMENT		In WAY4 it is possible to automatically attach/detach a custom fee to all main Services of a Service Package. To do so, follow the instructions below: • Register an additional Service Package. • In the additional Service Package, register a template Service (identical to the Service from the main Package). The ATTACHMENT tag is set for this template Service. • Configure a custom fee for the template Service.

Name	Value	Description
		• Configure the connection of the additional Service Package with the main Service Package. When the connection of the main Service Package and attached Package with the configured template Service is activated, a custom fee set up for the template Service will be attached to all Services of the main Package whose parameters correspond to those of the template Service. Note that only a custom fee is attached, a template Service is not attached to the main Package and is only used for comparison with the Service of the main Package to decide whether to attach a custom fee.
FEE_BASE	"TRANSACTION" "RECONCILIATION" "SETTLEMENT" "CUSTOM" <tag name> "TGTA" "TGTT" "SRCA" "SRCF" "SRCT" "SRCFD" "TGTFD" "FROM_BALANCE" "CR_TRANS" "DOC_TAG"	This tag determines the base amount for calculating a fee using one of the following tags: • "TRANSACTION" – tag specifying that the transaction amount will be used as the base amount. • "RECONCILIATION" – tag specifying that the reconciliation amount will be used as the base amount. • "SETTLEMENT" – tag specifying that the settlement amount will be used as the base amount. • "CUSTOM" – the value returned by the CUST_FEE_BASE procedure will be used to calculate the fee base. • <tag name> – the base amount and currency will be taken from the document's reason_details field from the tags <tag name>_CURR, <tag name>_AMOUNT. • "DOC_TAG" – used together with the FEE_BASE_TAG=<tag name> tag. The base amount and currency will be taken from the document's reason_details field from the <tag name>_CURR and <tag name>_AMOUNT tags. • "TGTA" – used to calculate a Custom Fee. The base for calculating the fee will be the amount and currency of the actual entry for the target contract (the amount from the t_amount field of the macrotransaction). • "TGTT" – used to calculate a Custom Fee. The base for calculating the fee will be the amount and currency of the actual entry for the target contract including the fee (the sum of the macrotransaction's t_amount and t_fee_amount). • "SRCA" – used to calculate a Custom Fee. The base for calculating the fee will be the amount and currency of the actual entry for the source contract (amount from the s_amount field of the macrotransaction). • "SRCF" – used to calculate a Custom Fee. The base for calculating the fee will be the amount and currency of the source contract fee specified in the S

Name	Value	Description
		Fee field of the macrotransaction (s_fee_amount). • "SRCT" – used to calculate a Custom Fee. The base for calculating the fee will be the amount and currency of the actual entry for the source contract including the fee (the sum of the macrotransaction's s_amount and s_fee_amount). • "SRCFD" – the base amount for calculating the fee is the source contract fee amount specified in the document (Source Fee Amount field (source_fee_amount)). The currency of the base amount for calculating a fee is taken from the _Source Fee Currency_ field of the document (source_fee_curr). • "TGTFD" – the base amount for calculating the fee is the target contract fee amount specified in the document (_Target Fee Amount_ field (target_fee_amount)). The currency of the base amount for calculating a fee is taken from the _Target Fee Currency_ field of the document (source_fee_curr). • "FROM_BALANCE" – used together with the tag FEE_BASE_TAG. When calculating a fee, the value of the balance type specified in the tag FEE_BASE_TAG=<balance type amount> is used as the base fee. • "CR_TRANS" – the base amount for calculating the fee is the transaction amount converted to the channel currency.
FEE_AMOUNT_TAG	<tag name>	Defines the name of the tag from which the amount for fee calculation will be taken from a document's Add Data (add_info) field. See also FEE_CURR_TAG.
FEE_CURR_TAG	<tag name>	Defines the name of the tag from which the currency for fee calculation will be taken from a document's Add Data (add_info) field. See also FEE_AMOUNT_TAG.
FEE_BASE_FROM	<fee code>	This tag allows a fee to be calculated as a percentage of the value of another fee (fee with a set code). A Service with this tag is used together with Services containing <CALC> tags. For example: • A custom fee must be configured (for example, Custom Fee1), for which a code is set in the Fee Code field (for example, 123). Set the <CALC> tag in this fee, meaning that this fee is used only for calculations (see the description above). • Another custom fee must be configured (for example, Custom Fee2), for which the value "FEE_BASE_FROM=123;" is specified in the Service Details field (where 123 is the code specified for the first custom fee (Custom Fee1)). The amount of Custom Fee2 will be calculated as a percentage from the amount of Custom Fee1. The tag can be set in a custom fee or in the main Service.

Name	Value	Description
		If the currencies of Custom Fee1 and Custom Fee 2 differ, when Custom Fee1 is converted to the currency of Custom Fee2, the FX Type, FX Rate Type and Direction parameters of Custom Fee1 are used (i.e., rules are used for converting the fee in the FEE_BASE_FROM tag).
FX_MARGIN		When this tag is set, a fee will be debited only during conversion and is the FX difference. The base amount for calculating the fee is the product of the Settlement Amount and the difference between the values of the Middle Rate and Buy/Sell Rate. When calculating the FX difference, errors may occur. The FX difference (FeeAmount) is calculated as the difference between the country's Central Bank (National Bank) rate on the transaction date and the actual FX rate of the transaction expressed in a local currency. FeeAmount=TransactionAmount(LocalAmount/TransactionAmount-CBRate). The actual FX rate of the transaction is calculated as the relationship of the LocalAmount to the TransactionAmount. The actual FX rate of the transaction may differ from the rate set for the FX scheme, due to rounding. This may lead to errors in the value of the FX difference (FeeAmount).
SRCF		The base amount for calculating a fee is the fee amount of the source contract specified in the S Fee (s_fee_amount) of the macrotransaction.
TGTF		The base amount for calculating a fee is the fee amount of the target contract specified in the T Fee (t_fee_amount) of the macrotransaction.
SRCFD		The base amount for calculating a fee is the fee amount of the source contract specified in the document (Source Fee Amount field (source_fee_amount)). The currency of the base amount for calculating a fee is taken from the document's Source Fee Currency field (source_fee_curr).
TGTFD		The base amount for calculating a fee is the fee amount of the target contract specified in the document (Target Fee Amount field (target_fee_amount)). The currency of the base amount for calculating a fee is taken from the document's Target Fee Currency field (target_fee_curr).
CUSTOM		The base amount for calculating a fee is the amount calculated using the CUST_FEE_BASE procedure. For detailed information, contact WAY4 vendor representatives.
ADD_FEE	<fee code>	Allows the value of another fee (fee with the set code) to be added to the fee amount (Custom Fee). For example: • A custom fee (Custom Fee1) is configured for which a code is specified in the Fee Code field (for example, 123). The fee percentage is 10%.

Name	Value	Description
		• Another custom fee is configured (Custom Fee2) for which the ADD_FEE=123; tag is specified. The fee percentage is 20%. With consideration of these settings, for Custom Fee2 the fee amount is calculated as follows: a fee in the amount of 20% of the main transaction amount is calculated, and the amount calculated according to Custom Fee1 (10% of the main transaction amount) is added to it. To add several fees to a custom fee, a fee code mask (<FeeCode>%) can be used as the tag value. If the currencies of Custom Fee1 and Custom Fee 2 differ, when Custom Fee1 is converted to the currency of Custom Fee2, the FX Type, FX Rate Type and Direction parameters of Custom Fee1 are used (i.e., rules are used for converting the fee in the ADD_FEE tag).
SUBTRACT_FEE	<fee code>	Allows a fee amount with a set code to be subtracted from the calculated fee amount (Custom Fee). A Service with this tag is used together with Services containing <CALC> tags. For example: • A custom fee must be configured (Custom Fee1), for which a code is set in the Fee Code field (for example, 123). Set the <CALC> tag in this fee, meaning that this fee is used only for calculations (see the description above). • Another custom fee must be configured (Custom Fee2), for which the value "SUBTRACT_FEE=123;" is specified in the Service Details field (where 123 is the code specified for the first custom fee (Custom Fee1)). The amount of Custom Fee2 will be calculated as the difference between Custom Fee2 and Custom Fee1. To subtract several fees from a custom fee, a fee code mask (<FeeCode>%) can be used as the tag value. If the currencies of Custom Fee1 and Custom Fee 2 differ, when Custom Fee1 is converted to the currency of Custom Fee2, the FX Type, FX Rate Type and Direction parameters of Custom Fee1 are used (i.e., rules are used for converting the fee in the SUBTRACT_FEE tag).
ROUND	"n" "-m" ">k" "<l" "=p"	Tag for rounding the result of fee calculation; using ROUND=<value>. Possible values of the tag: • "n" – rounding to 10-n ("0" –to a basic unit of currency, dollars, pounds), "1" – to a fractional currency, cents, pence), "2" – to units of fractional currency; in both cases it is assumed that the fractional currency portion of the amount is 2 decimal places. I.e. the amount of 20.78 USD will be rounded to 20.80 (to tenths) when the value is "1" (ROUND=1;) and when the value is "2" (ROUND=2;), it will remain the same. • "-m" – rounding to 10-m ("-1" – to tenths of the basic unit of currency, "-2" to

Name	Value	Description
		hundredths of the basic unit of currency). For example "ROUND=-2"; makes it possible to round 244.64 USD to 200 USD. • ">k" – rounding up to k fractional currency units. For example, when ROUND=>1000; the amount of 244.64 USD is rounded to 250.00 USD). When ROUND=>100;, the amount of 244.64 USD is rounded to 245.00 USD). Note. ROUND=>200 setup is used for rounding to even units of the main currency (for example, to 2/6/4/8 USD depending on the rounded amount). For example, when a fee rate is 3% and the transaction amount is 100 USD, the initially calculated fee amount of 3 USD will be rounded to 4 USD. In some cases, setup with the ROUND=>200 tag and 2% fee rate can be used. In this case, for example, a fee of 2 USD will be charged for each 100 USD transaction (a rounded fee of 2 USD will be charged for a transaction amount from 0.01 to 100 USD, for amounts from 100.01 USD to 200 USD, an additional 2 USD will be charged (i.e. the fee amount will be 4 USD)). • "<l" – rounding down to l fractional currency units. For example, when ROUND=<1000; the amount of 244.64 USD is rounded to 240.00 USD). When ROUND=<100;the amount of 244.64 USD is rounded to 244.00 USD). • "=p" – rounding according to a standard algorithm to fractional currency units. It is recommended to use this value to set up rounding to "p" units, hundredths of the main currency (for units, hundredths of dollars, pounds). I.e. it is recommended to use "100", "1000", etc. When ROUND==1000;, the amount of 125.65 is rounded to 130. When ROUND==100;, the amount of 125.65 is rounded to 126. When it is necessary to round to tenths, units of fractional currency (i.e. rounding the part of the amount "after the decimal"), use the ROUND=n; tag, see above.
OWN_VALUE_DAYS;		This tag is set for a custom fee and makes it possible to not use the settings of the main Service's Value Days field, and use the value of the custom fee's Fee Day field. If in the custom fee the Fee Day field is not filled in, the date of the contract account entry will be the current banking date.
MARKUP		Allows the amount of a custom fee to be included in the transaction amount. When making a transaction with this Service, in the ENTRY table (where technical entries are kept) one record is created where the total amount (transaction amount + fee amount) is reflected in the Amount field, and the "0" value is reflected in the Fee field.

Name	Value	Description
CALC_WHEN_CREDIT	"Y"	When the value is "Y", for in transactions on this Service for the account specified in the Service's Account Type field interest will be additionally accrued.
FEE_CONTRACT	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "SELF" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<имя тега> "LIAB_CATEGORY" "FROM_DECISION:<decision code>" "FROM_TARIFF:<tariff type code>"	Redefines the contract to which the fee is transferred (Fee Contract set in the Service). Possible options: • "BILLING" – the account contract from which settlement is made. • "LIABILITY" – higher ranking contract in the Liability hierarchy. • "TOP" – top contract in the hierarchy. • "BASE" – main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – contract specified in the Source Contract (source_contract) field of a document (see the form "Doc-Brief") • "DOC_TARGET" – contract specified in the Target Contract (target_contract) field of a document (see the form "Doc-Brief") • If the tag is not set or has the "FROM_SERVICE" value, the fee is transferred to the contract specified in the Fee Account field of the Service (fee_account) • "FROM_DOC" – contract is taken from the Add Data (add_info) field of a document according to the tag specified with the tag CONTRACT_TAG (in the same field) • "RELATED" – a related contract with the relation specified in the RELATION tag. • "SELF" – when this value is specified, a fee is transferred to the client contract account with the same code as the account of the bank contract specified in the Service. Note: If a contract set using the FEE_CONTRACT tag is not found, no fee will be charged (i.e. basic Service settings will not be applied). • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: FEE_CONTRACT=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration:

Name	Value	Description
		CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the FEE_CONTRACT=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R"), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. • "FROM_DECISION:<decision code>" – makes it possible to use decisions to redefine a fee contract. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes". • "FROM_TARIFF:<tariff type code>" – makes it possible to use a tariff to redefine a fee contract. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes". !"FROM_DECISION:<decision code>" and "FROM_TARIFF:<tariff type code>" values can be specified with the ":STRICT" postfix. For example, FEE_CONTRACT=FROM_DECISION:REV_CONTR_ISS:STRICT;. In this case, the bank contract number that is specified is used as it is defined in classifier/tariff parameters. By default (if the ":STRICT" postfix is not set) this contract number is used as a template. I.e. the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). The tag can be used together with the REDEF_FEE_ACC_CODE tag.

Name	Value	Description
CONTRACT_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY" "FROM_DECISION:<decision code>" "FROM_TARIFF:<tariff type code>"	Redefines the contract from which the fee is taken: • "BILLING" – the account contract from which settlement is made. • "LIABILITY" – higher ranking contract in the Liability hierarchy. • "TOP" – top contract in the hierarchy. • "BASE" – main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – contract specified in the Source Contract (source_contract) field of a document (see the form "Doc-Brief") • "DOC_TARGET" – contract specified in the Target Contract (target_contract) field of a document (see the form "Doc-Brief") • "FROM_SERVICE" – contract specified in the Service (fee_contract, fee_account) • "FROM_DOC" – contract is taken from the Add Data (add_info) field of a document according to the tag specified with the tag CONTRACT_TAG (in the same field) • "RELATED" – a related contract with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: CONTRACT_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category

Name	Value	Description
		"R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the CONTRACT_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. • "FROM_DECISION:<decision code>" – makes it possible to use decisions to redefine a fee contract. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes". • • "FROM_TARIFF:<tariff type code>" – makes it possible to use a tariff to redefine a fee contract. For more information, see the section "Redefining Contracts and Contract Accounts" of the document "Products and Contract Subtypes". !"FROM_DECISION:<decision code>" and "FROM_TARIFF:<tariff type code>" values can be specified with the ":STRICT" postfix. For example, CONTRACT_FOR=FROM_DECISION:REV_CONTR_ISS:STRICT;. In this case, the bank contract number that is specified is used as it is defined in classifier/tariff parameters. By default (if the ":STRICT" postfix is not set) this contract number is used as a template. I.e. the prefix of the corresponding financial institution is automatically added to it (prefix of the financial institution for the contract with which the operation is being made (Target or Source contract)). Note: If a contract set using the CONTRACT_FOR tag is not found, no fee will be charged (i.e. basic Service settings will not be applied). Note: The CONTRACT_FOR tag with the DOC_TARGET_NUMBER value is used to redefine the target contract of an interbank document (Fee Collection). When this tag is set, instead of being posted to a bank contract (NOSTRO contract, routing contract), a document is posted to a card contract registered in the issuer's database. The contract is determined according to the card number in the the document. This makes it possible to process an interbank document.

Name	Value	Description
CONTRACT_TAG	<tag name>	Sets the name of the tag redefining a fee contract in the document. The tag value is taken from the document's add_info field. Used together with the tags FEE_CONTRACT=FROM_DOC and CONTRACT_FOR=FROM_DOC for the way to redefine the FROM_DOC fee contract (see the tags FEE_CONTRACT and CONTRACT_FOR).
RELATION	<Relation Code>	The relation type code for a related contract when redefining a fee contract (see the tags FEE_CONTRACT and CONTRACT_FOR).
FROM_AUTH	SERVICE "SERVICE_TGT" "SERVICE_SRC" FEE_AMOUNT FX_RATE	This tag is used for a Package's main Services and is not used for a custom fee's Service (except the "FEE_AMOUNT" value). This tag specifies that if authorization was made on a Service for which this tag is specified, the financial document for this transaction will inherit the parameters set by the tag from the Service on which authorization was made. Tag values are: ▪ "SERVICE" – all Service parameters are inherited, meaning no Service will be searched for when posting the financial document. If a tariff was used when posting an authorization with this Service, a search will not be made for a tariff when posing the financial document - the tariff for authorisation will be used (i.e. parameters will be inherited from the corresponding tariff). When this value is set, both target contract Service parameters and source contract Service parameters are inherited. • "SERVICE_TGT" – target contract Service parameters are inherited. • "SERVICE_SRC" – source contract Service parameters are inherited. ▪ "FEE_AMOUNT" – only the fee amount is inherited. This value can be specified in the main Service or in a custom fee's Service. If the currency of the custom fee amount blocked by an authorization differs from the fee amount's currency, when making the corresponding entries, conversion is made according to the rules of the fee's Service. ▪ "FX_RATE" - FX rates at the time of authorization are inherited (regardless of the global parameter USE_AUTH_FX) All values can be set in the format: FROM_AUTH=SERVICE,FEE_AMOUNT,FX_RATE; The FROM_AUTH tag in a Service redefines the same tag in the Service Package or transaction subtype. I.e., first a search is made for the tag in the Service, then in the Service Package, and then in the transaction subtype. If the tag is found in the Service, the Service Package will not be searched. Recommends to use this tag in a transaction subtype and Service Package, as settings support at the level of certain Services is time-consuming.

Name	Value	Description
REV_ACC_CODE	<AccCode>	Allows redefinition of the account to which a reversal will be posted, when making a reversal on this Service.
DOC_TAG	<tag name>	Makes it possible to set conditions for selecting a Service depending on document parameters specified as tags in the Add Data and Reason Details fields. Several values, delimited by commas, can be specified as the value of the DOC_TAG_VALUE tag.
DOC_TAG_VALUE	<tag value> "PRESENT" "ABSENT"	"PRESENT" - if this value is set or the tag is not set, a check is made for the presence of the tag specified in the DOC_TAG tag (a check is made for the presence of the tag without a value). "ABSENT" - if this value is set, a check is made that the tag specified in DOC_TAG is present in the document (a check is made for the tag without a value).
SKIP_FI_HIERARCHY		When this tag is set, when determining a counterparty, the hierarchy of financial institutions is not considered. For example there are two preferred counterparties set up in the system – for the subordinate financial institution and the head financial institution. If a transaction is made on the card of a subordinate financial institution, by default the preferred conditions will be checked for the head financial institution. When this tag is set, the system will choose the preferred counterparty of the subordinate financial institution. This tag can be specified in the preferred counterparty (Custom Parms field).
FEE_PCNT_TAG	<tag name>	Specifies the name of the tag from which the fee percentage will be taken from the document's add_info field. !The tag should be only be used in special cases: for "Balance Transfer" transactions or transactions from external ("foreign") merchants for which a special interest rate is defined. Individual tariffs should be used for custom fees for "our" transactions.
FORCE_AMOUNT	<amount>	If the global parameter AUTH_USE_FORCE_AMOUNT has a value differing from the default one ("N"), if a request arrives with the tag PARTIAL_APPROVAL_SUPPORT=Y;, for this Service, the tags FORCE_AMOUNT and FORCE_CURR specified in the Service's Service Details field will be processed. Therefore, for a Service, the value of the amount that will be used in the authorization response will be set instead of the value that came in the authorization request.
FORCE_CURR	<Currency code>	See the description of the FORCE_AMOUNT tag.
MTR_FEE		Added ability to suppress creation of a separate document for a fee charged when a limiter is activated. The macrotransaction for charging the fee is

Name	Value	Description
		created under the financial document for the transaction. The tag can also be specified in the transaction subtype.
HIDE_BALANCE		This tag is used to suppress sending balance information in a response to an authorization request. This tag can also be specified in the transaction subtype in the Fee Algorithm Options (fee_algorithm) field.
INVOICE_ACTION	INSTALMENT EARLY_REPAYMENT	This tag is used when working with the WAY4 Instalments module. An instalment schedule is generated when a financial document is processed using the Service with the corresponding marker. This marker is the INVOICE_ACTION=INSTALMENT; tag in the Service's Service Details field. When this marker is found, an instalment scheme is searched for. For early repayment of a plan on a document (i.e. at the same time funds are transferred) the Service for early repayment must contain the INVOICE_ACTION=EARLY_REPAYMENT tag. The module is delivered according to a separate agreement with the WAY4 vendor.
REASON_CODE	<dispute document reason code>	Allows redefinition of the reason code for generating a dispute document according to payment system regulations.
OWN_CURR	"Y" "N"	The tag makes it possible to set conditions for selecting a Service for transactions in the contract currency/not in the contract currency: • When the value is "Y", the Service will be used for the transaction if the transaction currency (by default, the Settlement Currency in the document is used) corresponds to the contract currency. • When the value is "N", the Service will be used for the transaction if the transaction currency (by default, the Settlement Currency in the document is used) differs from the contract currency. The document currency type that is compared with the contract currency can be set with the OWN_CURR_TYPE tag.
ACTIVITY_FOR	"S" "T" "B"	When this parameter is set, the date of an operation on this Service will be recorded under the contract using the ACTIVITY_DATE tag in the ext_data field of the Acnt_contract table: • "S" – source Service • "T" – target Service • "B" – both target and source Service • When the ACTIVITY_FOR tag is set, the following should be noted: The ACTIVITY_FOR tag should only be used for card contract transactions. I.e. for subtypes/Services with a card contract as the source or target. Depending on this, the value of the ACTIVITY_FOR tag should be set to "S"

Name	Value	Description
		or "T", respectively. • The ACTIVITY_FOR tag with the "B" value can only be set for transactions between two card contracts. • If the ACTIVITY_FOR tag with the "B" value is set for transactions where the second participant is a device contract or account contract (bank contract), transactions for these contracts may be processed in different parallel threads, leading to contracts being blocked and document posting errors.
FEE_AMOUNT_IN_LOCAL_CUR RENCY		This tag makes it possible to avoid rounding errors in conversion when charging a fee set as a fixed amount in a local currency from an account in a foreign currency, and to pay the set fixed amount in the local currency to the corresponding bank revenue account.
GROUP_CODE	<group code>	GROUP_CODE=<group code>; - tag for grouping Services for usage limiters (see the section document "Usage Limiters"). If this tag is specified in a Service/Services (for example, GROUP_CODE=XXX;), and "XXX;" value (for example "XXX,YYY,…ZZZ;") is specified in the Serv Group Code field of the limiter using this Service, the limiter will activate when a transaction is made on any Service with this group code. This functionality works for a limiter configured in the contract on whose Service the transaction is made as well as for a limiter configured in a higher ranking contract (Main/Sub, Liability), if this limiter is checked according to a Usage Scenario. For example, it is possible to specify a group code for the limiter of a higher-ranking Liability contract, then the limiter will activate when transactions are made on Services of the group of subordinate contracts if the corresponding group code is set for this group's Services.
FORCE_SIC_TO		The tag makes it possible to redefine the SIC code for outgoing documents (performed in the Service of the corresponding device contract or routing contract).
IF_CS_TYPE	<Classifier type code>	This tag sets the type of classifier checked. Used together with the IF_CS_VALUE, IF_NOT_CS_VALUE, IF_CS_TYPE_FOR tags. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers". It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
IF_CS_VALUE	<CS_STATUS_VALUE.CODE 1>,	A Service that is appropriate according to all other parameters for processing a transaction will be used if clients and their contracts correspond to the set

Name	Value	Description
	<CS_STATUS_VALUE.CODE 2>,…,<CS_STATUS_VALUE.C ODE N>	value of this classifier. Several codes separated by commas may be specified as the value of this tag. Used together with the IF_CS_TYPE tag. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers". It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
IF_NOT_CS_VALUE	<CS_STATUS_VALUE.CODE 1>, <CS_STATUS_VALUE.CODE 2>,…,<CS_STATUS_VALUE.C ODE N>	A Service that is appropriate according to all other parameters for processing a transaction will not be used if clients and their contracts correspond to the set value of this classifier. Several codes separated by commas may be specified as the value of this tag. Used together with the IF_CS_TYPE tag. For more information, see the section "Executing Actions Depending on Classifier Values" of the document "WAY4™ Client and Contract Classifiers". It is possible to set a check of several classifiers for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. See the description of the IF_CS_NUMB tag.
STATUS_CATEGORY	"V" "I" "D"	A Service meeting all other parameters for processing a transaction will be selected if the contract is in the status with the specified category (the value of the Is Valid field of the "Contract Statuses" form "Full → Configuration Setup → Contract Types → Contract Statuses"): • "V" – Valid • "D" – Decline • "I" – Invalid
USE_OVL;		The tag allows of the contract amount available to be calculated when processing partial approval with consideration of the amount set using the "Overdraft" limiter.
INST_PLAN_TO_ADDENDUM;		When this tag is set, the instalment plan calculated according to this Service is transformed into a tag string and put in Addendum Doc. This tag is used to send the instalment plan calculated according to an authorisation document to the POS. The tag is used if the instalment plan is calculated when processing an authorisation document. The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. For more information, see the document "Instalment Loans in WAY4".
FINAL_AUTH_SAVING_PERIOD	<positive whole integer>	Redefines the global parameter of the same name. For more information, see the document "WAY4™ Global Parameters"

Name	Value	Description
REDEF_FEE_ACC_CODE	<account type code> "FROM_DECISION:<decision code>" "FROM_TARIFF:<tariff type code>"	This parameter makes it possible to redefine the type code of the account to which a custom fee is transferred. Used together with the FEE_CONTRACT tag. Can be used together with the REDEF_FEE_ACC_CURR tag. If the contract does not have an account with the code specified in REDEF_FEE_ACC_CODE, the fee will be posted to the "Fee Account" account of the "Fee Contract" contract according to Service settings. This will be reflected in the corresponding warning and error messages. The tag is used for not to create in the current contract a technical account to be specified in the Service parameters. For example, when an account type is used that is absent in the current contract (and that is present in the contract defined using the CONTRACT_FOR tag).
REDEF_ACC_CODE	<account type code> "FROM_DECISION:<decision code>" "FROM_TARIFF:<tariff type code>"	This parameter makes it possible to redefine the type code of the account from which a custom fee is charged. Used together with the CONTRACT_FOR tag. The tag is used for not to create in the current contract a technical account to be specified in the Service parameters. For example, when an account type is used that is absent in the current contract (and that is present in the contract defined using the CONTRACT_FOR tag).
FEE_BASE_ADD_MARKUP		The tag makes it possible to add a custom fee with the MARKUP code to the base amount for fee calculation (which in turn is determined by the FEE_BASE tag). If the fee currency and base amount currency differ, when converting the custom fee to the base amount currency, the custom fee's FX Type, FX Rate Type and Direction parameters are used. The FEE_BASE_ADD_MARKUP; tag can be used both together with the FEE_BASE tag and independently. The FEE_BASE_ADD_MARKUP; tag can be used on the custom fee level and on the Service level. For example, if the FEE_BASE_ADD_MARKUP; tag is set in a Service, a custom fee with the MARKUP code will be added to the Transaction Amount of the macrotransaction. If both tags are used, for example, FEE_BASE=RECONCILIATION;FEE_BASE_ADD_MARKUP; WAY4 takes Reconcil Amount as the basis and adds a custom fee with the MARKUP code, converted to the currency in which the bank provides transaction information to the payment system (Reconcil Currency).
CHECK_DATA_PRIOR		The tag allows selection of a tariff with the "Service" role" with consideration of priority (the due_period_grace field of the tariff_data table) in a domain

Name	Value	Description
		hierachy. If the tag is not set, the tariff is selected with consideration of domain priority (the priority of this tariff type in different domains in the hierarchy is not considered).
FEE_MIN_MODE		Allows a custom fee to be charged only when the amount of the transaction on the Service is positive. The tag is used together with the FEE_BASE=SETTLEMENT; tag. The "SETTLEMENT" value of the FEE_BASE take makes it possible to consider a fee taking the sign into account (if this value is not set, the fee is always considered as a positive value, even if it is calculated on the basis of a negative transaction/document amount) and the FEE_MIN_MODE=POSITIVE; tag makes it possible to charge only positive fee amount values. These tags are used in particular to configure the fee charged when increasing a credit limit. In this case, the custom fee is configured under the Service for processing the credit limit. When setting a credit limit for a contract online, the limit is set with the fee already considered (subtracted) (if the fee is debited from an account with the "Is Am Av" marker, the fee is not blocked). When proccessing the macrotransaction, fee blocking is removed.
IF_PARM	<tagged parameter name> CLIENT.SHORT_NAME CLIENT.FIRST_NAME CLIENT.LAST_NAME CLIENT.COMPANY_NAME CLIENT.AGE	This tag sets a checked classifier (tagged parameter) for a contract. Used together with the IF_PARM_VALUE tag. For more information, see the section "Classifiers without a Fixed List of Values" of the document "WAY4™ Client and Contract Classifiers". A number of client parameters can be checked. To do so, the following values can be specified as the tag's value: CLIENT.SHORT_NAME – check the Short Name field in the client form. CLIENT.FIRST_NAME – check the First Name field in the client form. CLIENT.LAST_NAME – check the Last Name field in the client form. CLIENT.COMPANY_NAME – check the Company Name field in the client form (place of work) CLIENT.AGE – client age. Determined according to date of birth (Date of Birth field) and the system date (i.e. age at the time of the request is determined). These client attributes do not have to be registered as contract custom parameters. It is possible to set a check of several parameters for a contract by using the IF_PARM_NUMB, IF_PARM<N>, IF_PARM_VALUE<N>, IF_PARM_FOR<N> tags. See the description of the IF_PARM_NUMB tag.
IF_PARM_VALUE	"<tagged parameter value>" "NOT_EMPTY" "EMPTY"	A Service conforming to all other parameters for processing an operation will be selected if clients and their contracts correspond to the set classifier value. Used together with the IF_PARM tag. The IF_PARM_VALUE=NOT_EMPTY;

Name	Value	Description
	"LIST_WITH:\<list of values separated by commas\>" "BETWEEN\<value\> AND \<value\>"	tag is used to check for the existence of a parameter (the tag set using the IF_PARM tag) without checking its value. When the IF_PARM_VALUE=EMPTY; tag is set, the Service will be selected if the specified parameter doesn't have a value. Note that with this value, the Service will also be selected if the parameter is not present in the contract. When "IF_PARM_VALUE=LIST_WITH:\<list of values separated by commas\>" is set, the Service will be selected if even one value from the list (LIST_WITH) matches that set in the contract. When "IF_PARM_VALUE=BETWEEN\<value\> AND \<value\>" the Service will be selected if the value in the contract falls in the range set by the tag (for example, IF_PARM_VALUE=BETWEEN40 AND 50;). The value must be set with spaces, as shown in the example. It is possible to set a check of several parameters for a contract by using the IF_PARM_NUMB, IF_PARM\<N\>, IF_PARM_VALUE\<N\>, IF_PARM_FOR\<N\> tags. See the description of the IF_PARM_NUMB tag.
ZERO_BAL_TYPE	\<balance type code\>	A Service will only be selected if the value of the balance specified in the tag is equal to zero. The tag can be used together with the ZERO_BAL_TYPE_FOR tag.
NON_ZERO_BAL_TYPE	\<balance type code\>	A Service will only be selected if the value of the balance specified in the tag is not equal to zero. The tag can be used together with the NON_ZERO_BAL_TYPE_FOR tag.
ALLOW_ZERO_ENTRIES;		The tag is used to enable due normalisation for a null amount. See the description of the ALLOW_ZERO_DUE tag for account template.
CURR_FROM_FEE_CURR		The tag is set for a custom fee. The tag makes it possible to charge a custom fee strictly in the set currency - the currency set on the tariff level or in the fee currency, no tariff is set. The setting works if the Account Curr field is not filled in for the custom fee. The tag can be used for Misc Fees in the same way as Custom Fee.
HOLD_PERIOD_TYPE	"WRK"	The HOLD_PERIOD_TYPE=WRK; tag makes it possible to specify the length of time funds will be held when processing an authorisation request in banking working days. The number of days (calendar) for holding is set in the global parameter AUTH_SAVING_PERIOD. This tag is set on the Service level (SERVICE table) in the Service Details field (service_details). In addition, the tag CALENDAR_TYPE=\<CalendarTypeName\>; can be used to specify the calendar type name of the calendar that will be used to define the hold period. If a calendar type is not specified, the default business

Name	Value	Description
		calendar will be used. The CALENDAR_TYPE tag is also set on the Service level (SERVICE table) in the Service Details field (service_details).
CALENDAR_TYPE	<Calendar Type Name>	The tag CALENDAR_TYPE=<CalendarTypeName>; can be used to specify the calendar type name of the calendar that will be used to define the hold period when processing an authorisation request (see the description of the HOLD_PERIOD_TYPE tag). If the calendar type is not specified, the default business calendar will be used.
USE_CONTRA_TRF		The tag is set for a custom fee: - under a card contract's Service and makes it possible to use a tariff set for a merchant contract (for a counterparty contract) to calculate a custom fee recorded in the card's account. - under a device contract's Service and makes it possible to use a tariff set for a card contract (for a counterparty contract) to calculate a custom fee recorded in the merchant's account. The tariff can be configured in the Product or in a specific contract.
BAL_TYPE	<balance type code>	Tags are used to limit the total amount of a custom fee charged for a period. Set for a custom fee. At the present time the functionality is only used for credit custom fees (with "Credit" in the Fee Dir field). The value of the BAL_TYPE tag is the balance type code of the balance used to record (accumulate) a fee for a period (the period is set using the HISTORY_MODE parameter).
MAX_TOTAL_AMOUNT_PARM	<contract parameter code>	Note that if the BAL_TYPE tag is set in the main Service and the same account is used to record the fee for the main Service and Custom Fee, a code for the fee must be specified in the Fee Code field and the BAL_TYPE tag must either not be set or its value must differ from the value in the Service. Otherwise the Custom Fee will be included in the amount of turnover for the balance type of the main Service. The <MAX_TOTAL_AMOUNT_PARM=<contract parameter code> and the corresponding parameter on the contract leve are used to set a limit on the maximum total amount of a fee that can be charged for a period. The value set with the contract parameter is compared with the amount of the balance BAL_TYPE=<balance type code>. If the balance amount exceeds the maximum fee amount, no fee is charged. If the balance amount is less than this limit, but when the fee is charged, the balance amount exceeds the maximum fee amount, the full size of the fee will not be charged (the fee amount consists of the difference between the balance amount and the maximum total amount of the fee). A limit can be set for the maximum total fee amount so that no fee is charged

Name	Value	Description
		(not accumulated) for a specific balance type. To do so, the MAX_TOTAL_BAL_TYPE=<balance type code> tag is used together with the MAX_TOTAL_AMOUNT_PARM=<contract parameter code> tag. When charging a fee, the amount of the specified balance type is compared with the maximum total fee amount and the fee does not change the balance amount.
SOURCE_CHANNEL TARGET_CHANNEL	"ON_US" "OFF_US" <comma-delimited list of channel codes>	Tags make it possible to configure additional conditions for selecting a Service depending on source and target channels.
USE_TRANS_AMOUNT	"Y" "N" "F"	The tag is used to determine how to calculate an amount debited from "on-us" card accounts when a financial document imported from the payment system is posted, as well as the amount of funds blocked when an authorisation request is processed: • When the value is "F", the transaction amount is blocked and debited if the card has an account in the transaction currency. • When the value is "Y", the transaction amount is always blocked and debited. • If the tag is not set or is set to "N", the settlement amount will be debited. The tag redefines the tag of the same name set in the transaction subtype, Product, or Service Package level and by the global parameter of the same name. For more information, see the section "Calculating the Debited Amount (USE_TRANS_AMOUNT) tag" of the document "Financial Institutions".
AUTH_USE_DOMESTIC	"Y" "C" "A" "N"	The tag can be used to redefine the value of the tag of the same name set on the Product, Service Package, financial institution, transaction subtype level, as well as the value of the global parameter of the same name (for more details, see the document "WAY4™ Global Parameters".
TOTAL_BAL_TYPE	<balance type code>	The TOTAL_BAL_TYPE tag is used to set the code of the balance type that is checked. This balance is compared with the amount set in the tariff with the Threshold role (the checked tariff is set using the LIMIT_AMOUNT=<tariff type code>; tag). When a transaction is made, the settlement amount is added to the balance with the specified code and compared with the maximum available balance set with the tariff. If the balance type is greater than the amount set in the tariff, the transaction is rejected. Used to configure prepaid card top-up. For more information, see the section

Name	Value	Description
		"Configuring Prepaid Card Top-Up" of the "WAY4™ Advanced Tariff Management" document.
OWN_CURR_TYPE	"TRANS_CURR" "RECONS_CURR"	The tag defines the document currency (Settlement Currency, Transaction Currency, Reconciliation Currency) that will be compared with the contract currency when processing the OWN_CURR tag. If the tag is not set, the document's Settlement Currency will be used by default.
CHECK_BAL		The tag makes it possible to not check available funds when making a transaction, when the following conditions are observed: 1. The CHECK_BAL=N; tag is set in the Service 2. The CHECK_BAL=N; tag is set in the template of the account used for the transaction. 3. The Is Am Av field value of the account used for the transaction is "No"
IF_CURRENCY	<tag name> TRANS_CURR SETTL_CURR RECONS_CURR CONTRACT <tag or contract custom parameter> <classifier code> <three-digit numeric currency code or currency name>	This tag sets the document currency that must be checked. Possible values: • Document tag containing the currency code. • TRANS_CURR – transaction currency from the document. • SETTL_CURR – settlement currency from the document. • RECONS_CURR – reconciliation currency from the document (in the currency in which the source bank provides transaction information to the payment system). • CONTRACT – contract currency. • Tag or contract custom parameter with a currency code. A contract's custom parameter can be registered as a "Primary" classifier. •Code of the classifier with a currency code. •Explicit specification of currency – the tag value may be a code or the name of a certain currency. Used together with the IF_CURRENCY_VALUE, IF_CURRENCY_RULE, IF_CURRENCY_FOR tags.
IF_CURRENCY_VALUE	EMPTY NOT_EMPTY USE_IN_BANK <currency code1>, <currency code2>,…,<currency codeN> <currency name1>, <currency name2>,…,<currency nameN> <tag name>	A Service suitable according to all other parameters for processing a transaction will be selected if the document currency specified with the IF_CURRENCY tag corresponds to parameters set in the IF_CURRENCY_VALUE tag. Possible values: • EMPTY – this currency is not specified. • NOT_EMPTY – this currency is specified. • USE_IN_BANK – this currency has the Use in Bank parameter value of "Yes" (i.e. for this currency, conversion can be performed in WAY4 and the rate of the currency to the financial institution's local currency can be entered during the

Name	Value	Description
		daily opening procedure). • A code or name of a certain currency or several currencies, separated by commas can be used as the tag value. • Tag or contract custom parameter containing a currency code or list of currencies separated by commas can be used as the tag value. A contract custom parameter can be registered as a "Primary" classifier.
IF_CURRENCY_RULE	ACC_SCHEME EQUAL NOT_EQUAL NOT_IN_LIST	To set additional conditions for checking a currency, use the IF_CURRENCY_VALUE tag together with the IF_CURRENCY_RULE tag: • To check the currency set in the IF_CURRENCY for correspondence to the Accounting Scheme it is necessary to set the tags IF_CURRENCY_RULE=ACC_SCHEME;IF_CURRENCY_VALUE=<list of account codes separated by commas>;. If a list of account codes is set, the currency is searched for among the specified Accounting Scheme account templates. If a list of accounts is not set, the currency is searched for among all Accounting Scheme account templates. • To check a currency for correspondence (or non-correspondence) to another document currency, it is necessary to set the tags IF_CURRENCY_RULE=EQUAL; or IF_CURRENCY_RULE=NOT_EQUAL; the value of IF_CURRENCY_VALUE is also set as the value of the IF_CURRENCY tag. • When IF_CURRENCY_RULE=NOT_IN_LIST; is specified, a check will be made that the currency is not included in the list of currencies specified explicitly or through the parameter in the tag IF_CURRENCY_VALUE.
SAVE_FEE_TAG	<tag name>	The tag set using the SAVE_FEE_TAG tag is used when saving the amount of a miscellaneous fee in a document. If the tag is set, specify the <SaveFeeTag>_AMOUNT tag as the value of the global parameter AMOUNT_FORMAT_FOR_DOC_TAG. If SAVE_FEE_TAG=<>; is not set, the values of the global parameter will be searched for the <FeeCode>_AMOUNT tag. For more information, see the description of the SAVE_FEE tag.
SAVE_FEE;		The tag is set in a Custom Fee and allows the fee amount to be saved in document tags. By default, the value is saved using the following tags: <FeeCode>_AMOUNT, <FeeCode>_CURR, <FeeCode>_ID. A tariff identifier, if set, is specified in the <FeeCode>_ID tag. A different code can be used instead of the Fee Code to save custom fee values, specifying it using the SAFE_FEE_TAG=<>; tag. The value saved in the <FeeCode/SaveFeeTag>_AMOUNT is not stored in ISO format, therefore the <FeeCode>_AMOUNT and <SaveFeeTag>_AMOUNT

Name	Value	Description
		tags must be added to the list of tags in the global parameter AMOUNT_FORMAT_FOR_DOC_TAG. This tag can be set in a custom fee with the <CALC>; tag. This tag can be used to separately charge fees for transactions with different acquirer accounts. A calculated fee value is saved in document tags even if it is equal to "0". A fee value is not saved if the Service is not suitable according to check results (classifiers, other tags, preferred counterparties, etc.)
DATE_MODE	"BILLING" "TRANS_DATE"	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag sets the mode for calculating the effective date for each instalment from a plan: • "BILLING" – the start date of the corresponding billing cycle is the effective date for an instalment. • "TRANS_DATE" – the effective date of an instalment is calculated in months from the date of the original transaction; i.e. if the transaction date was 14 February, the effective date will be calculated first from 14 February, then from 14 March, 14 April, etc. For more information, see the document "Instalment Loans in WAY4".
INST_FIN_ACTION	"CHECK" "ACTIVATE" "CREATE_INACTIVE" "CREATE_ACTIVE" "CREATE_PREVIEW" "ACTIVATE_RECALC"	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag redefines the value of the same global parameter. For more information, see the section "Configuring Financial Document Processing (Automatic Creation of a Plan)" of the document "Instalment Loans in WAY4".
INST_AUTH_ACTION	"CHECK" "CREATE_INACTIVE" "CREATE_PREVIEW" "SIMULATED"	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag redefines the value of the same global parameter. For more information, see the section "Configuring Authorisation Document Processing (Automatic Creation of a Plan)" of the document "Instalment Loans in WAY4".
INST_AUTHCHECK_ACTION	"CHECK" "CREATE_INACTIVE" "CREATE_PREVIEW" "SIMULATED"	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag redefines the value of the same global parameter. For more information, see the section "Configuring "AuthCheck" Preauthorisation Document Processing (Automatic Creation of a Plan)" of the document "Instalment Loans in WAY4".

Name	Value	Description
INST_IRATE	<fee interest rate>	The tag is used in the WAY4 Instalments module. The tag is used to set the interest rate for calculating a fee. The tag redefines settings in the instalment scheme for an "Interest" fee (Calc Scheme field value). I.e. the fee size is taken from the tag and the calculation scheme from instalment scheme fee settings. If the INST_IRATE tag is set, and there is no "Interest" fee in the instalment scheme, this fee percentage will be calculated according to the "Interest 360" scheme. For more information, see the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
PORT_FRATE	<interest rate>	The tag is used in the WAY4 Instalments module. The tag sets the size of the fee (as a percentage of the transaction amount) for one instalment.The tag redefines instalment scheme settings for "Annual Fee" and "Flat Fee" fees (Calc Scheme value). For more information, see the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
PORT_FTOTAL	<amount>	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag sets a fixed fee amount in each instalment (see the description of the Base Amount field in the instalment scheme). The tag redefines instalment scheme settings for "Annual Fee" and "Flat Fee" fees (Calc Scheme field value). For more information, see the document "Instalment Loans in WAY4".
INST_FWAIVE_N	<number of portions	The tag is used in the WAY4 Instalments module. The module is supplied according to an additional agreement with the WAY4 vendor. The tag sets the number of instalments for which no fee is charged. This tag is used together with the PORT_FRATE tag. For more information, see the document "Instalment Loans in WAY4".
INST_AMNT	<amount of one instalment>	The tag is used in the WAY4 Instalments module. The tag sets the amount of one instalment in an instalment plan. For more information, see the document "Instalment Loans in WAY4". The module is supplied according to an additional agreement with the WAY4 vendor.
TRANS_FX		The tag sets an additional condition for selecting a Service when posting a transaction (document). When the tag is set, the Service will only be used if the transaction currency differs from the settlement currency, i.e. if conversion is performed.
SIC_LIST	<SIC1>,<SIC2>,…<SICN>	The tag sets an additional condition for selecting a Service when posting a transaction (document). The Service is used if a transaction is made at certain

Name	Value	Description
		merchants (of the specified type). The tag sets a list of SICs (MCC), separated by commas, for which this Service can be used.
EXC_SIC_LIST	<SIC1>,<SIC2>,…<SICN>	The tag sets an additional condition for selecting a Service when posting a transaction (document). If the list includes the document's SIC (MCC), the Service will not be selected. Codes are separated by commas.
MATCH_REVERSE_ONLINE	<time interval>	The tag specifies the time interval (in hours) relative to the time of the original transaction, after which all reversals (reversing the original transaction) will be rejected.
DUE_DATE_TAG	<tag name>	The DUE_DATE_TAG tag determines the document tag that will be set in a document's Add Info field and contains a date in "YYYYMMDD" format. The tag sets the Due Date that will be used as Due Date for records in the ITEM table, in which entries are record that were generated by transactions with this Service. I.e. the tag is used to redefine the Due Date for an account's amount accrued in a specific record in the ITEM table. The account template for which these entries are made must have the "Value Due Date" value of the Due Type field.
CHECK_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Conditions for selecting a Service IF_PARM, IF_ACTIVE_EVNT, IF_CS set in the Service are checked for the contract specified in the CHECK_FOR tag. The Service meeting all other parameters for processing the transaction will be selected if the check is successful. Possible values for the tag: • "BILLING" – the check is made for the account contract. • "LIABILITY" – the check is made for the higher ranking contract in the Liability hierarchy. • "TOP" – the check is made for the top contract in the hierarchy. • "BASE" – the check is made for the main contract in a "Main/Sub" hierarchy, with which this contract is related (with which transactions are made). • "DOC_SOURCE" – the check is made for the contract specified in the Source Contract (source_contract) field of the document (see the form "Doc-Brief") • "DOC_TARGET" – the check is made for the contract specified in the Target Contract (target_contract) field of the document (see the form "Doc-Brief") •"FROM_SERVICE" - the check is made for the contract specified in the Service (fee_contract, fee_account field). • "FROM_DOC" – the check is made for the contract from the Add Data (add_info) field of the document according to the tag specified with the CONTRACT_TAG tag (in the same field) • "RELATED" – the check is made for a related contract with the relation

Name	Value	Description
		specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: CHECK_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the CHECK_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. Note: if the contract set with the CONTRACT_FOR tag is not found, a check will be made for the original contract.

Name	Value	Description
REDEF_FEE_ACC_CURR	SETTL_CURR TRANS_CURR RECONS_CURR <document tag> FEE_CURR	The tag makes it possible to redefine the currency of the account specified in the Fee Account field (value of the account to which the fee is transferred). The REDEF_FEE_ACC_CURR tag can be used independently or together with the REDEF_FEE_ACC_CODE and/or FEE_CONTRACT tags. Tag values: ▪ SETTL_CURR – the settlement currency from the document will be used as the account currency. ▪ TRANS_CURR – the transaction currency from the document will be used as the account currency ▪ RECONS_CURR – the reconciliation currency from the document (in the currency in which the source bank provides transaction information to the payment system) will be used as the account currency ▪ <document tag> – document tag containing the currency code. ▪ FEE_CURR - the currency set in the Service (Service's Fee Curr field) will be used as the account currency. Fee Currency can be set directly in a Service or defined through a tariff.
DATE_EVENT_TYPE_MODE	FORTNIGHTLY SEMIANNUAL	Provides additional functionality to determine the frequency for charging a Misc fee with the "Weekly" and "Yearly" values in the Charge Event field: ▪ FORTNIGHTLY – the fee is charged once every two weeks. When the tag has this value, the Value Days field works as follows: when a value from 1 to 7 is specified, the fee is charged on the specific day of each oddly-numbered week; when a value from 8 to 14 is specified (where "8" = "Monday", etc.) the fee is charged on the specified day of each evenly-numbered week. ▪ SEMIANNUAL – the fee is charged semi-annuallly.
IF_PARM_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	The IF_PARM_FOR tag is used together with the IF_PARM/IF_PARM_VALUE tags to redefine the contract for which these checks are made. • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the document's Source Contract (source_contract) field (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract specified in the document's Target Contract (target_contract) field (see the "Doc-Brief" form). • "FROM_DOC" – the contract is taken from the document's Add Data (add_info) field, according to the tag specified with the CONTRACT_TAG tag

Name	Value	Description
		(in the same field). • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_PARM_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL ;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the IF_PARM_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. It is possible to set a check of several parameters for a contract by using the IF_PARM_NUMB, IF_PARM<N>, IF_PARM_VALUE<N>, IF_PARM_FOR<N> tags. See the description of the IF_PARM_NUMB tag.

Name	Value	Description
IF_CS_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	The IF_CS_TYPE_FOR tag is used together with IF_CS group tags to redefine the contract for which these checks are made. • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – from a higher-ranking contract in a "Liability" hierarchy. • "TOP" – from the top contract in a hierarchy. • "BASE" – from the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the document's Source Contract (source_contract) field (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract specified in the document's Target Contract (target_contract) field (see the "Doc-Brief" form). • "FROM_DOC" – the contract is taken from the Add Data (add_info) field. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_CS_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the

Name	Value	Description
		IF_CS_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. It is possible to set a check of several parameters for a contract by using the IF_CS_NUMB, IF_CS_TYPE<N>, IF_CS_VALUE<N>, IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N>. See the description of the IF_CS_NUMB tag.
IF_CURRENCY_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the IF_CURRENCY tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the Source *Contract field* of the document (source_contract) of the document (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract in the Target *Contract* field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's Add *Data field* (add_info) according to the tag specified using the CONTRACT_TAG tag in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_CURRENCY_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be

Name	Value	Description
		selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the IF_CURRENCY_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
ZERO_BAL_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<ta	Redefines the contract for which a check is made using the ZERO_BAL_TYPE tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the Source *Contract field* of the document (source_contract) of the document (see the"Doc-Brief" form). • "DOC_TARGET" – for the contract in the Target Contract *field* (target_contract) of the document (see the "Doc-Brief" form).

Name	Value	Description
	g name> "LIAB_CATEGORY"	• "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's Add Data *field* (add_info) according to the tag specified using the CONTRACT_TAG tag in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: ZERO_BAL_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the ZERO_BAL_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for

Name	Value	Description
		example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
NON_ZERO_BAL_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the NON_ZERO_BAL_TYPE tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the Source Contract field of the document (source_contract) of the document (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract in the Target Contract field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's Add Data field (add_info) according to the tag specified using the CONTRACT_TAG tag in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: NON_ZERO_BAL_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag

Name	Value	Description
		• "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" – "Only Check Balance" category. For example, when the NON_ZERO_BAL_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
MAX_BAL_TYPE	<balance type code>	The MAX_BAL_TYPE tag sets the code of the balance type being checked. This balance is compared with the amount set in a tariff with the Threshold role (the checked tariff is set using the LIMIT_AMOUNT=<tariff type code>; tag). If the balance type is greater than the amount set in the tariff, the transaction is declined. Tariff setup is described in the document "WAY4 Advanced Tariff Management".
MIN_BAL_TYPE	<balance type code>	The MIN_BAL_TYPE tag sets the code of the balance type being checked. This balance is compared with the amount set in a tariff with the Threshold role (the checked tariff is set using the LIMIT_AMOUNT=<tariff type code>; tag). If the balance type is less than the amount set in the tariff, the transaction is declined. Tariff setup is described in the document "WAY4 Advanced Tariff Management".
LIMIT_AMOUNT	<tariff type code>	Sets the code of the tariff type with the Threshold role used to set the threshold amount. This amount is compared with the balance set using one of the following tags: TOTAL_BAL_TYPE, MAX_BAL_TYPE, MIN_BAL_TYPE (tag is set in the same field).
IF_CS_NUMB	<number of additional checks>	Sets the number of additional checks for the IF_CS_TYPE tag. The tag is used together with the IF_CS_TYPE<N>, IF_CS_VALUE<N>,

Name	Value	Description
		IF_NOT_CS_VALUE<N>, IF_CS_TYPE_FOR<N> tags. I.e. by default the IF_CS_TYPE tag is checked. If the IF_CS_NUMB=<number of additional checks>; tag is set, the specified number of IF_CS_TYPE<N> tags is checked. For example, if IF_CS_NUMB=3;, the IF_CS_TYPE1, IF_CS_TYPE2, IF_CS_TYPE3 tags are checked in addition to the IF_CS_TYPE tag.
IF_PARM_NUMB	<number of additional checks>	Sets the number of additional checks for the IF_PARM tag. The tag is used together with the IF_PARM<N>, IF_PARM_VALUE<N>, IF_PARM_FOR<N> tags. I.e. by default the IF_PARM tag is checked. If the IF_PARM_NUMB=<number of additional checks>; tag is set, the specified number of IF_PARM<N> tags is checked. For example, when IF_PARM_NUMB=3;, the IF_PARM1, IF_PARM2, IF_PARM3 tags are checked in addition to the IF_PARM tag.
IF_COMPARE	EQUAL NOT_EQUAL	This tag expands the area of checks made using IF_CS, and IF_PARM tags. The IF_COMPARE tag is used to set up conditions for executing actions in WAY4 depending on whether parameter values match. The tag makes it possible to compare different parameters for the same contract or for different contracts (for example, Source and Target contract parameters), as well as document and contract parameters. A pair of entities is compared, for example "Classifier/Classifier", "Parameter/Parameter", "Classifier/Parameter" or "Classifier/Document tag". The IF_COMPARE tag sets the rule for comparison: • "EQUAL" – checks that the first value matches the second value, or that both are empty. • "NOT_EQUAL" – checks that the first value DOES NOT match the second value. The tag is used together with the IF_COMPARE_FIRST and IF_COMPARE_SECOND tags. A check of several parameter pairs for a contract can be set with the IF_COMPARE_NUMB, IF_COMPARE<N>, IF_COMPARE_FIRST<N>, IF_COMPARE_SECOND<N> tags. See the description of the IF_COMPARE_NUMB tag.
IF_COMPARE_FIRST	PARM_<Code> TAG_<Code> CS_<Code>	The IF_COMPARE_FIRST tag together with the IF_COMPARE_SECOND tag sets parameters for comparison. Possible values: • PARM_<Code> – the code of the custom parameter for comparison is set in this format. • TAG_<Code> – the code of the tagged parameter for comparison is set in this format. • CS_<Code> – the code of the classifier for comparison is set in this format.

Name	Value	Description
		There may be no value for the IF_COMPARE_SECOND tag. In this case, the same parameter/classifier specified in the IF_COMPARE_FIRST tag will be checked. The tag is used together with the IF_COMPARE tag. A check of several parameter pairs for a contract can be set with the IF_COMPARE_NUMB, IF_COMPARE<N>, IF_COMPARE_FIRST<N>, IF_COMPARE_SECOND<N> tags. See the description of the IF_COMPARE_NUMB tag.
IF_COMPARE_SECOND	PARM_<Code> TAG_<Code> CS_<Code>	The IF_COMPARE_SECOND tag together with the IF_COMPARE_FIRST tag sets parameters for comparison. Possible values: • PARM_<Code> – the code of the custom parameter for comparison is set in this format. • TAG_<Code> – the code of the tagged parameter for comparison is set in this format. • CS_<Code> – the code of the classifier for comparison is set in this format. There may be no value for the IF_COMPARE_SECOND tag. In this case, *the* same parameter/classifier specified in the IF_COMPARE_FIRST tag will be checked. The tag is used together with the IF_COMPARE tag.
IF_COMPARE_FIRST_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the IF_COMPARE_FIRST tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the *Source Contract* field of the document (source_contract) of the document (see the"Doc-Brief" form). • "DOC_TARGET" – for the contract in the *Target Contract* field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's *Add Data* field (add_info) according to the tag specified using the CONTRACT_TAG tag (in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role

Name	Value	Description
		code> tag in this field; for example: IF_COMPARE_FIRST_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYM ENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the IF_COMPARE_FIRST_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. A check of several parameter pairs for a contract can be set with the IF_COMPARE_NUMB, IF_COMPARE<N>, IF_COMPARE_FIRST<N>, IF_COMPARE_SECOND<N> tags. See the description of the IF_COMPARE_NUMB tag.

Name	Value	Description
IF_COMPARE_SECOND_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "DOC_TARGET_NUMBER" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the IF_COMPARE_SECOND tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the Source Contract field of the document (source_contract) of the document (see the "Doc-Brief" form). • "DOC_TARGET" – for the contract in the Target Contract field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's Add Data field (add_info) according to the tag specified using the CONTRACT_TAG tag in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_COMPARE_SECOND_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the

Name	Value	Description
		LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the IF_COMPARE_SECOND_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. A check of several parameter pairs for a contract can be set with the IF_COMPARE_NUMB, IF_COMPARE<N>, IF_COMPARE_FIRST<N>, IF_COMPARE_SECOND<N> tags. See the description of the IF_COMPARE_NUMB tag.
TOTAL_BAL_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the TOTAL_BAL_TYPE tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for a higher-ranking contract in a "Liability" hierarchy. • "TOP" – for the top contract in a hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract in the Source Contract field of the document (source_contract) of the document (see the"Doc-Brief" form). • "DOC_TARGET" – for the contract in the Target Contract field (target_contract) of the document (see the "Doc-Brief" form). • "FROM_SERVICE" – for the contract specified in the Service (fee_contract, fee_account fields). • "FROM_DOC" – the contract is taken from the document's Add Data field (add_info) according to the tag specified using the CONTRACT_TAG tag (in the same field) • "RELATED" – related contract, with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example:

Name	Value	Description
		TOTAL_BAL_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the TOTAL_BAL_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
CONV_CHANNEL		The tag is used to configure conversion of Fee Min, Fee Max, Fee Base amounts from an arbitrary currency to an account currency at payment system rates. Required settings: - Set the CONV_CHANNEL; tag in the Service.

Name	Value	Description
		- A special composite FX type must be configured CR_<Buy>;<Sell>;<Middle> (with "Yes" in the Is Internal field; Full\Configuration Setup\Accounting Setup\FX Types). - This FX type must be set in the Service. - The BASE_AMOUNT_FX_RATE_TYPE=SERVICE; tag should be set in Service details (or specify "SERVICE" for the global parameter of the same name). If the tag/parameter is not set, conversion will be made at the "Middle" rate.
IF_COMPARE_NUMB	<number of additional checks>	The tag sets the number of additional checks for IF_COMPARE_FIRST, IF_COMPARE_SECOND parameter pairs. I.e. if the IF_COMPARE tag is set, the IF_COMPARE_FIRST and IF_COMPARE_SECOND tags are checked. if IF_COMPARE_NUMB=<number of additional checks>; is set, the specified number of IF_COMPARE<N>, IF_COMPARE_FIRST<N>, IF_COMPARE_SECOND<N> tag sets will be checked.
BASE_AMOUNT_FX_RATE_TYPE	"SERVICE" "M" "B"	The tag is set in the main Service and makes it possible to define rules for converting Fee Base, Fee Min, and Fee Max fees with settings in a particular Service. The tag redefines the same global parameter. See the section "BASE_AMOUNT_FX_RATE_TYPE" of the document "WAY4 Global Parameters". "SERVICE" - the value is used if the global parameter of the same name is not set and rules for converting the main amount of the transaction for the Service do not differ from rules for converting Fee Min, Fee Max, and Fee Base fees. When this value is specified Fee Base, Fee Min, and Fee Max fees are converted at the rate set in the Service's FX Rate Type field. "M" (Middle) - conversion at the middle rate. "B" (BuySell) - conversion at the buy-sell rate. The "M" and "B" values are used when rules for converting the main amount of a transaction for a Service differ from rules for converting Fee Min, Fee Max, and Fee Base fees.
INST_PAYIN	"I" "B"	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. Possible values: • "I" – only generation of an instalment plan is possible (default value) • "B" – standard payment and generation of an instalment plan is possible. See the section "Configuring Functionality to Pay for Transactions using Simulated Instalment Plans" of the document "Instalment Loans in WAY4".

Name	Value	Description
MIN_BAL_TYPE_SIGN	"+" "-"	The tags must be used when checking minimum/maximum balances that can have positive or negative values. For example, Service balances are kept as positive, but in posting reversals, may change the sign. The tags are used together with the MIN_BAL_TYPE and MAX_BAL_TYPE tags.
MAX_BAL_TYPE_SIGN	"+" "-"	If the MIN_BAL_TYPE_SIGN=+; tag is set, a negative balance will be considered as a null balance. If the MIN_BAL_TYPE_SIGN=-; tag is set, a positive balance will be considered as a null balance.
MAX_FEE	SRCF SRCA SRCT TGTF TGTA TGTT TRANSACTION RECONCILIATION SETTLEMENT <tag name> "SRCFD" "TGTFD" "FROM_BALANCE" "CR_TRANS" "DOC_TAG"	The tag is set in a custom fee. The tag makes it possible to limit the maximum fee amount by the amount from a macrotransaction or from a document. Tag values: • SRCF – the Custom Fee amount will be limited by the amount of the source contract's fee specified in the S Fee field of the macrotransaction (s_fee_amount). • SRCA – the Custom Fee amount will be limited by the amount of the source contract's entry (macrotransaction s_amount). • SRCT– the Custom Fee amount will be limited by the amount of the source contract's entry including the fee (macrotransaction s_amount and s_fee_amount). • TGTF – the Custom Fee amount will be limited by the amount of the target contract's fee specified in the macrotransaction's T Fee field (t_fee_amount). • TGTA – the Custom Fee amount will be limited by the amount of the target contract's entry (macrotransaction's t_amount). • TGTT – the Custom Fee amount will be limited by the amount of the target contract's entry including the fee (macrotransaction's t_amount and t_fee_amount). • "TRANSACTION" - the tag specifies that an amount in the transaction currency is used as the base amount. •"RECONCILIATION" - the tag specifies that an amount in the currency in which the source bank provides transaction information to the payment system (Reconciliation Amount) will be used as the base amount. • "SETTLEMENT" - the tag specifies that an amount in the settlement currency (Settlement Amount) will be used as the base amount. • <tag name> - the base amount and currency will be taken from the <tag name>_CURR and <tag name>_AMOUNT tags in the document's ADD_INFO field. • "SRCFD" – the base amount is the source contract fee amount specified in the document (Source Fee Amount field (source_fee_amount)). The currency

Name	Value	Description
		of the base amount for calculating the fee is taken from the document's Source Fee Currency field (source_fee_curr). • "TGTFD" – the base amount is the target contract fee amount specified in the document (Target Fee Amount field (target_fee_amount)). The currency of the base amount for calculating the fee is taken from the document'sTarget Fee Currency field (target_fee_curr). • "FROM_BALANCE" – used together with the FEE_BASE_TAG tag. The value of the balance type specified in the FEE_BASE_TAG=<balance type code> tag is used as the base amount. • "CR_TRANS" – the base amount is the Transaction Amount, converted to the account currency at the payment system's rate (Channel Currency). • "DOC_TAG" – used together with the MAX_FEE_TAG=<tag name> tag. The base amount and currency will be taken from the <tag name>_CURR and <tag name>_AMOUNT tags in the document's reason_details field. If a Custom Fee with the MAX_FEE tag is less than the limit amount, the full Custom Fee amount is charged. If the amount of the Custom Fee with the MAX_FEE tag is more than the limit amount, the fee amount is reduced correspondingly (for Custom Fee an amount equal to the limit amount is charged).
INST_SIM_SAVE	"F" "P" "M" "N"	The tag redefines the global parameter of the same name. Used to define how the data of simulated instalment plans are saved. Possible values: • "F" – a plan's full data are saved (default value). • "P" – plans are partially saved. When this value is specified, the plan's main record is saved and detailed information for the plan's components - data for Principal and Fee amounts, interest rate used in calculation, etc., (see the "Subtotals" form; "Instalments → Invoices & Instalments for Contracts → [Simulated] → [Subtotals]). Information about instalments is not saved in this mode. • "M" – a plan's minimum data are saved. When this value is specified, the plan's main record is saved (without detailed information about plan components and without information about instalments) • "N" – plans are not saved. For more information, see the document "Instalment Loans in WAY4™". The module requires a separate license from the WAY4 vendor.
EVENT_BEFORE	<Event Type Code>	The tag is set in a Misc service. The tag is used together with the EVENT_BEFORE_DAYS tag. The tags make it possible to open an Event several days before a fee is to be charged. For example, to send the client a

Name	Value	Description
		notification. The value of the fee amount according to the modulo is saved in Event parameters. The tag can be set in a transaction subtype. The same Event can be used for notification about any Misc fees, since information about an expected fee goes into the Event's tags when it opens.
EVENT_BEFORE_DAYS	<number of days>	The tag is set in a Misc service. The tag is used for fees with the values "Monthly", "Quarterly", "Yearly", "Billing Date", "Use Contract Date" in the Charge Event field. The tag is used together with the EVENT_BEFORE tag. The tags make it possible to open an Event several days before a fee is to be charged. For example, to send the client a notification. The amount of the fee for the notification is calculated when the Event opens. This amount may differ from the amount that will be debited. The amount may change if contract tariffs change or if the base amount for calculating the fee changes (for example if the fee is calculated as a percentage of the balance amount; the balance at the time of notification may differ from the balance when the fee is charged). The value of the fee amount according to the modulo is saved in Event parameters. The tag can be set in a transaction subtype. Note that for annual fees, the EVENT_BEFORE_DAYS tag can be specified ONLY in the transaction subtype.
TOTAL_BAL_TYPE_SIGN	"+"	When checking a balance type set with the TOTAL_BAL_TYPE tag if the tag TOTAL_BAL_TYPE_SIGN=+; is set, the balance amount will be taken with the corresponding sign (with the "+" sign if the balance is positive and with the "-" sign if it is negative). If the tag is not set, the balance amount is taken without consideration of the sign (by a module). Use to set up top-up of prepaid cards. See the section "Configuring Prepaid Card Top-Up" of the document "WAY4™ Advanced Tariff Management".
MAX_FEE_FROM	<Custom Fee code>	The tag is set in a Custom Fee and allows the fee amount to be limited by the amount of another Custom Fee.
MIN_FEE	SRCF SRCA SRCT TGTF TGTA TGTT	The tag is set in a Custom Fee. The tag makes it possible to limit the maximum fee amount by the amount from a macrotransaction or from a document. The list of the tag's values is identical to the list of MAX_FEE tag values (see the description above).

Name	Value	Description
	TRANSACTION RECONCILIATION SETTLEMENT <tag name> "SRCFD" "TGTFD" "FROM_BALANCE" "CR_TRANS" "DOC_TAG"	
MIN_FEE_FROM	<Custom Fee code>	The tag is set in a Custom Fee. The tag makes it possible to calculate the minimum fee amount as a percentage of another Custom Fee's value (fee with a specific code). Note. If the amount of Custom Fee2 is calculated as a percentage of the amount of Custom Fee1 and the currencies of Custom Fee1 and Custom Fee2 differ, when converting the amount of Custom Fee1 to the currency of Custom Fee2, the FX Type, FX Rate Type and Direction parameters of Custom Fee1 are used (i.e. fee conversion rules set in the MIN_FEE_FROM tag are used).
MAX_FEE_FROM	<Custom Fee code>	The tag is set in a Custom Fee. The tag makes it possible to calculate the maximum fee amount as a percentage of another Custom Fee's value (fee with a specific code). Note. If the amount of Custom Fee2 is calculated as a percentage of the amount of Custom Fee1 and the currencies of Custom Fee1 and Custom Fee2 differ, when converting the amount of Custom Fee1 to the currency of Custom Fee2, the FX Type, FX Rate Type and Direction parameters of Custom Fee1 are used (i.e. fee conversion rules set in the MAX_FEE_FROM tag are used).
MIN_FEE_AMOUNT_TAG	<tag name>	The tag is set in a Custom Fee. Defines the name of the tag from which the amount for calculating the minimum fee will be taken in the document's Add Data (add_info) field. See also MIN_FEE_CURR_TAG. The tags are used if document tags for storing the amount and currency have names that do not contain the suffixes _AMOUNT -for the amount, and _CURR - for the currency. Otherwise, use the tags MIN_FEE=DOC_TAG;MIN_FEE_TAG=<tag name>.
MIN_FEE_CURR_TAG	<tag name>	The tag is set in a Custom Fee. Defines the name of the tag from which the amount for calculating the minimum fee will be taken in the document's Add

Name	Value	Description
		Data (add_info) field. See also MIN_FEE_AMOUNT_TAG. The tags are used if document tags for storing the amount and currency have names that do not contain the suffixes _AMOUNT -for the amount, and _CURR - for the currency. Otherwise, use the tags MIN_FEE=DOC_TAG;MIN_FEE_TAG=<tag name>.
MAX_FEE_AMOUNT_TAG	<tag name>	The tag is set in a Custom Fee. Defines the name of the tag from which the amount for calculating the maximum fee will be taken in the document's Add Data (add_info) field. See also MAX_FEE_CURR_TAG. The tags are used if document tags for storing the amount and currency have names that do not contain the suffixes _AMOUNT -for the amount, and _CURR - for the currency. Otherwise, use the tags MAX_FEE=DOC_TAG;MAX_FEE_TAG=<tag name>."
MAX_FEE_CURR_TAG		The tag is set in a Custom Fee. Defines the name of the tag from which the amount for calculating the maximum fee will be taken in the document's Add Data (add_info) field. See also MAX_FEE_AMOUNT_TAG. The tags are used if document tags for storing the amount and currency have names that do not contain the suffixes _AMOUNT -for the amount, and _CURR - for the currency. Otherwise, use the tags MAX_FEE=DOC_TAG;MAX_FEE_TAG=<tag name>."
FEE_BASE_TAG	<tag name>	Used together with the "FROM_BALANCE" and "FROM_TAG" values of the FEE_BASE tag. See the description above.
MAX_FEE_TAG		The tag is set in a Custom Fee. Used together with the "FROM_BALANCE" and "FROM_TAG" values of the MAX_FEE tag. See the description above.
MIN_FEE_TAG		The tag is set in a Custom Fee. Used together with the "FROM_BALANCE" and "FROM_TAG" values of the MIN_FEE tag. See the description above.
FEE_INTERBRANCH	ORDER	Special standing payment order for redistributing fees in interbranch operations (with "Interbranch" in the Date Event field) can be used for documents created when limiters were activated (for Misc services). To do so, set the FEE_INTERBRANCH=ORDER; tag in the corresponding Misc service or in the additional fee for the Misc service. For this functionality to work correctly, to make balance inquiries and statement requests, the value of the global parameter

Name	Value	Description
		LOG_ALL_OPERATIONS must be set to "Y". The tag can be set in the main Service or in the Service of a custom fee.
INST_FROM_TRANS_DATE		The tag makes it possible when posting a document for the Service to calculate the fee for an instalment plan from the transaction date (by default, a fee is calculated from the date the financial document is posted). If the billing cycle to which the transaction date belongs is already closed, system behaviour depends on the value of the instalment scheme's Dates Mode/BIlling Mode field. If the date calculation mode is based on a billing cycle, dates are calculated from the start of the current billing cycle. If calculation starts from the transaction date ("By Trans Date" mode) and no instalment plan shift is set, the first portion can become effective immediately when the financial document is posted, and may already be overdue.
CONTR_TYPES	<list of contract type codes>	The tag is set in an additional Service (Service of an additional Package with the ATTACHMENT tag). The tag is used to limit the list of contract types to which a custom fee for this Service can be attached. Specify a list of contract type codes, separated by commas, as the value (the value of the RBS Code field in the contract type form, for example "Device Contract Type" (Full → Configuration Setup → Contract Types → Device Contract Types)). For example, the CONTR_TYPES=AT,IM,PO; tag value in a Service means that it can be attached to three types of contract: ATM, Imprinter, POS. !If in the handbook ("Device Contract Type" in the example) a code is specified with spaces at the beginning or end, this code should be specified without spaces as the value of the CONTR_TYPES tag. Otherwise, an error message will be generated when approving the Package. The tag is used together with the ANY_CONTR_TYPE tag and the ATTACHMENT tag. If the tag is absent, the Service can be attacked to a Package with any type of contract (see the description of the ANY_CONTR_TYPE tag).
CONTRA_TYPES	<list of counterparty contract type codes>	The CONTRA_TYPES=<value> tag is set in an additional Service (Service of an additional Package with the ATTACHMENT tag). The tag makes it possible to attach this Service's custom fees to the Services of the main package with different types of counterparty contract (Service's CONTRA_TYPE field). I.e. one Service can be configured in an additional Package with the ATTACHMENT tag and a custom fee can be attached to Services with different type of counterparty contracts. Specify a list of contract type codes, separated by commas, as the value (the value of the RBS Code field in the contract type form, for example "Card

Name	Value	Description
		Contract Type" (Full → Configuration Setup → Contract Types → Card Contract Types"). For example, the CONTRA_TYPES=CM,EC,PC,VC; tag value in a Target Service means that a custom fee will be attached to all Services with the values Cirrus/Maestro, MasterCard, Electron/Plus, VISA Cards in the Source Type field. !If in the handbook ("Card Contract Type" in the example) a code is specified with spaces at the beginning or end, this code should be specified without spaces as the value of the CONTRA_TYPES tag. Otherwise, an error message will be generated when approving the Package. In this case, when searching for and attaching this Service, the values of the Target Type/Source Type (CONTRA_TYPE) fields and the Transaction Type of the Service are dynamically substituted with the parameters of the main Service Package's corresponding Services.
ATTACH_MAIN	Y	The ATTACH_MAIN=Y; tag makes it possible to use a Service from a Package with the ATTACHMENT tag not only to attach custom fees, but to attack the Service itself. The tag is set in the attached Service. The Service is added to the main package, and then custom fees set up for it are attached. For attached Services, the values of the Contract Type and Transaction Type fields are dynamically substituted with the parameters of the main Service Package's corresponding Services. The ATTACH_MAIN=Y tag is only available for Services with the CONTRA_TYPE=<value>; tag set.
EXC_SIC_GROUP_LIST	<SIC-group code1>,<SIC-group code2>, ...,<SIC-group codeN>	If a document's SIC code (MCC code) is included in this group, the Service will not be selected. Group codes (the value of the Group Code field in the "SIC Groups" form (Full → Configuration Setup → Main Tables → SIC Groups)) are separated by commas.
IF_ACTIVE_EVNT	<Event type code>	A Service that is suitable according to all other parameters for processing the transaction will only be selected if an Event of the type specified in the tag (with the specified code) is open.
IF_INACTIVE_EVNT	<Event type code>	A Service that is suitable according to all other parameters for processing the transaction will only be selected if an Event of the type specified in the tag (with the specified code) is closed.
REQUEST_CATEGORY	<Request Category>	The tag makes it possible to redefine the Request Category for recording transactions made with this Service/Custom Fee (with the tag set) in balance types. For example, transactions with the "Advice" and "Adjustment" category made with a Service with the REQUEST_CATEGORY=R; tag are recorded in

Name	Value	Description
		the corresponding balance types as reversals (as transactions with *the "*Reversal" category). The tag can be set in the transaction subtype. Note that a transaction will be recorded with the specified *category for* all related balances affected by the entry for this service, including balances in account templates.
MAX_BAL_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the MAX_BAL_TYPE_FOR tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for the higher ranking contract in the Liability hierarchy. • "TOP" – top contract in the hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the Source Contract *(source*_contract) field of a document (see the form "Doc-Brief") • "DOC_TARGET" – for the contract specified in the Target Contract *(target*_contract) field of a document (see the form "Doc-Brief") • "FROM_SERVICE" - for the contract specified in the Service (fee_contract, fee_account fields) • "FROM_DOC" – contract is taken from the Add Data *(add*_info) field of a document according to the tag specified with the tag CONTRACT_TAG (in the same field) • "RELATED" – a related contract with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: MAX_BAL_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT _LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration. For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL=

Name	Value	Description
		LEVEL=LEVEL1; tag. The check is mad for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the MAX_BAL_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
MIN_BAL_TYPE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_SERVICE" "FROM_DOC" "RELATED" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=<tag name> "LIAB_CATEGORY"	Redefines the contract for which a check is made using the MIN_BAL_TYPE_FOR tag. Possible values: • "BILLING" – for the account contract from which settlement is made. • "LIABILITY" – for the higher ranking contract in the Liability hierarchy. • "TOP" – for the top contract in the hierarchy. • "BASE" – for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – for the contract specified in the Source Contract (source_contract) field of a document (see the form "Doc-Brief") • "DOC_TARGET" – for the contract specified in the Target Contract (target_contract) field of a document (see the form "Doc-Brief") • "FROM_SERVICE" - for the contract specified in the Service (fee_contract, fee_account fields) • "FROM_DOC" – contract is taken from the Add Data (add_info) field of a document according to the tag specified with the tag CONTRACT_TAG (in the same field) • "RELATED" – a related contract with the relation specified in the RELATION tag. • "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role

Name	Value	Description
		code> tag in this field; for example: MIN_BAL_TYPE_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_ LEVEL;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: FOr example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as the LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the MIN_BAL_TYPE_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories.
PREAUTH_SAVING_PERIOD		Makes it possible to redefine the global parameter PREAUTH_SAVING_PERIOD. See the parameter section "PREAUTH_SAVING_PERIOD"

Name	Value	Description
STOP_AFTER_CARD_EXPIRE		The tag regulates charging recurring (Misc) fees when a Card Expire date is reached. When this tag is set, a fee is not charged if the date for charging it is greater than the date in the card contract's CARD_EXPIRE field (ACNT_CONTRACT table). If the Card Expire date falls on a weekend/holiday, a fee is only charged if the date for charging it is less than or equal to the Card Expire date. The tag can be set in a transaction subtype.
TARIFF_FROM_TRANS_DATE	<without a value> <number of days>	The tag makes it possible to use the calendar date of a transaction (Transaction Date) to search for a tariff (instead of the banking date used by default). The parameter's default value (i.e. if the tag is set without a value) is "10". Transaction Date is used to search for a tariff if this date falls in the interval defined as 10 days before and after the transaction's Posting Date. The number of days can be changed. To do so, specify the tag with a value (TARIFF_FROM_TRANS_DATE=<number of days>). The TARIFF_FROM_TRANS_DATE tag can be set in a tariff.
IF_DATE	"CLIENT.BIRTH_DATE" "CLIENT.DATE_EXPIRE" "CLIENT.DATE_OPEN" "CLIENT.ADD_DATE_01" "CLIENT.ADD_DATE_02" "CONTRACT.FIRST_ACTIVITY_DATE" "CONTRACT.DATE_OPEN" "CONTRACT.LAST_BILLING" "CONTRACT.NEXT_BILLING" "CONTRACT.DATE_EXPIRE" "CONTRACT.CARD_EXPIRE"	IF_DATE group tags make it possible to configure actions (in this case, selection of a Service) depending on contract/client dates. Contract/client dates can be compared with other dates (contract, client dates, system date, current banking date). Periods between dates can be checked according to various conditions. The IF_DATE tag sets the code of the date being checked, or the code of the date being compared with another date. Date codes can be contract/client dates specified in the following contract table (CONTRACT) or client table (CLIENT) fields: • "CLIENT.BIRTH_DATE" – a check is made according to the BIRTH_DATE field of the client's record • "CLIENT.DATE_EXPIRE" – a check is made according to the DATE_EXPIRE field of the client's record • "CLIENT.DATE_OPEN" – a check is made according to the DATE_OPEN field of the client's record (date on which the client was registered in the DB) • "CLIENT.ADD_DATE_01" – a check is made according to the ADD_DATE_01 field of the client's record • "CLIENT.ADD_DATE_02" – a check is made according to the ADD_DATE_02 field of the client's record • "CONTRACT.FIRST_ACTIVITY_DATE" – a check is made according to the FIRST_ACTIVITY_DATE field of the contract's record • "CONTRACT.DATE_OPEN" – a check is made according to the DATE_OPEN field of the contract's record

Name	Value	Description
		• "CONTRACT.LAST_BILLING" – a check is made according to the LAST_BILLING field of the contract's record • "CONTRACT.NEXT_BILLING"– a check is made according to the NEXT_BILLING field of the contract's record • "CONTRACT.DATE_EXPIRE" – a check is made according to the DATE_EXPIRE field of the contract's record • "CONTRACT.CARD_EXPIRE" – a check is made according to the CARD_EXPIRE field of the contract's record The tag is used together with the IF_DATE_VALUE and IF_DATE_FOR tags. A check of several dates for a client/contract can be set using the IF_DATE_NUMB, IF_DATE\<N\>, IF_DATE_VALUE\<N\> and IF_DATE_FOR\<N\> tags. See the description of the IF_DATE_NUMB tag.

IF_DATE_VALUE	"EMPTY"
	"NOT_EMPTY"
	\<Rule\>\<Base Date Code\>\<Period Rule\>
		The IF_DATE_VALUE tag sets rules for checking the date specified with the IF_DATE tag.
		Tag values:
		• "EMPTY" – the condition is met if the field for the date set with the IF_DATE field is not filled in.
		• "NOT_EMPTY" – the condition is met if the field for the date set with the IF_DATE field is filled in. The field value is not checked.
		• The tag value can be composite and set in the following format (components are listed without delimiters): IF_DATE_VALUE=\<Rule\>\<Base Date Code\>\<Period Rule\>;, where:
		  ▪ \<Rule\> – comparison rule. Possible values:
		  ♦ "\<" – less than
		  ♦ "\>" – greater than
		  ♦ "\<=" – less than or equal to
		  ♦ "\>=" – greater than or equal to
		  ♦ If a comparison rule (\<Rule\>) is not set, the date is checked for matching (for the "=" value).
		  ▪ \<Base Date Code\> – code of the date with which the date from the IF_DATE tag is compared. Possible values:
		  ♦ Values that are described for the IF_DATE tag can be specified (i.e. dates from the contract or client table).
		  ♦ "LDATE" – current banking date of the contract's financial institution
		  ♦ "DB_DATE" – system (calendar) data (the system date is used without time)
		  ▪ \<Period Rule\> – period that should be added or subtracted from the \<Base Date Code\> date. Set in the format \<sign\> \<number of units \>\<period unit of

Name	Value	Description
		measurement>, where ♦ \<sign> – "+" or "-" (sign for adding or subtracting the specified period) ♦ \<number of units> – number of days/months/years (see below) ♦ \<period unit of measurement> – "D" (days), "M" (months), "Y" (years). Only one value can be set (i.e. one type of period is set in the tag). ♦ If the period \<Period Rule> is not set, the two specified dates are compared according to the condition \<Rule>. The tag is used together with the IF_DATE and IF_DATE_FOR tags. A check of several dates for a client/contract can be set using the IF_DATE_NUMB, IF_DATE\<N>, IF_DATE_VALUE\<N>, and IF_DATE_FOR\<N> tags. See the description of the IF_DATE_NUMB tag. Examples of checks: Check of whether a client is already 40 years old: IF_DATE=CLIENT.BIRTH_DATE;IF_DATE_VALUE=<=DB_DATE-40Y; Check that a card has not expired: IF_DATE=CONTRACT.CARDS_EXPIRE;IF_DATE_VALUE=>=DB_DATE; Check that a card expires after a month: IF_DATE=CONTRACT.CARD_EXPIRE;IF_DATE_VALUE=<=DB_DATE+1M;
IF_DATE_FOR	"BILLING" "LIABILITY" "TOP" "BASE" "DOC_SOURCE" "DOC_TARGET" "FROM_DOC" "CONTRACT_ROLE" CONTRACT_ROLE_PARM=\<tag name> "LIAB_CATEGORY"	The IF_DATE_FOR tag is used together with the IF_DATE/IF_DATE_VALUE tags to redefine the contract/client for which the check is being made. Note. If the IF_DATE tag is used to check a date from the client table, first a search is made for a contract according to the IF_DATE_FOR tag, and then this date is checked for the corresponding client. Possible values for the IF_DATE_FOR tag: • "BILLING" – the check is made for the account contract used for settlement. • "LIABILITY" – the check is made for the higher ranking contract in the Liability hierarchy. • "TOP" – the check is made for the top contract in the hierarchy. • "BASE" – the check is made for the main contract in a "Main/Sub" hierarchy, with which this contract is related. • "DOC_SOURCE" – the check is made for the contract specified in the *Source Contract* (source_contract) field of the document (see the form "Doc-Brief") • "DOC_TARGET" – the check is made for the contract specified in the *Target Contract* (target_contract) field of the document (see the form "Doc-Brief") • "FROM_DOC" – the check is made for the contract from the *Add Data* (add_info) field of the document according to the tag specified with the CONTRACT_TAG tag (in the same field)

Name	Value	Description
		• "CONTRACT_ROLE" – for this value, specify the CONTRACT_ROLE=<role code> tag in this field; for example: IF_PARM_FOR=CONTRACT_ROLE;CONTRACT_ROLE=PAYMENT_LEVEL ;. In this case, a search will be made for a contract with the CONTRACT_ROLE=<role code>; tag upward in a Liability hierarchy (in our example, a search for the CONTRACT_ROLE=PAYMENT_LEVEL tag). The check is made for the contract with the specified tag. If no contract with this tag is found in the Liability contract, the top contract in the hierarchy will be selected. • When a search for a contract must be made in a Liability hierarchy according to the value of an arbitrary tag, the CONTRACT_ROLE_PARM=<tag name>; tag must be additionally used in the configuration: For example, when the following settings are specified in the configuration: CONTRACT_ROLE=LEVEL1;CONTRACT_ROLE_PARM=LEVEL; WAY4 will search upward in the Liability hierarchy for a contract with the LEVEL= LEVEL=LEVEL1; tag. The check is made for the contract with the specified tag • "LIAB_CATEGORY" – in this case, a search for a contract will be made upward in a Liability hierarchy within the category/categories set in this field using the LIAB_CATEGORY=<category1>,<category2> tag. Liability category codes, separated by commas, are specified as th LIAB_CATEGORY tag value: "Y" – "Full Liability" category "N" – "Affiliated" category "R" – "Reporting" category "A" - "Only Check Balance" category. For example, when the IF_PARM_FOR=LIAB_CATEGORY;LIAB_CATEGORY=Y,N; tags are specified, a search is made for the top contract in a "Liability" hierarchy, starting from the current contract within the "Full Liability" ("Y") and "Affiliated" ("N") categories. If a contract has no higher-ranking contracts, it will be used. If a higher-ranking contract belongs to another LIAB_CATEGORY, for example "Reporting" ("R")), no move up the hierarchy will be made. I.e. a contract from the middle of the Liability hierarchy will be used, the top one in in the "Full Liability" ("Y") and "Affiliated" ("N") categories. A check of several dates for a client/contract can be specified using the IF_DATE_NUMB, IF_DATE<N>, IF_DATE_VALUE<N>, and IF_DATE_FOR<N> tags. See the description of the IF_DATE_NUMB tag.

Name	Value	Description
IF_DATE_NUMB	<number of additional checks>	The tag sets the number of additional checks for the IF_DATE tag. The tag is used together with the IF_DATE<N>, IF_DATE_VALUE<N>, and IF_DATE_FOR<N> tags. I.e. by default, the IF_DATE tag is checked. If the IF_DATE_NUMB=<number of additional checks>; tag is set, the specified number of IF_DATE<N> tags is checked. For example, when IF_DATE_NUMB=3; in addition to the IF_DATE tag, the IF_DATE1, IF_DATE2, and IF_DATE3 tags are checked.
Tags in the *Fee Code* field of a Service		
MARKUP		Allows the amount of a custom fee to be included in the transaction amount. When making a transaction with this Service, in the ENTRY table (where technical entries are kept) one record is created where the total amount (transaction amount + fee amount) is reflected in the Amount field, and the "0" value is reflected in the Fee field.