Operation Manual

WAY4™ Dictionaries

03.50.30

28.02.2020

Contents

Introduction

The document is intended for bank or processing centre employees responsible for configuring WAY4 and describes product creation and configuration.

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

- "WAY4™ Service Packages"
- "Daily Procedures"
- "Financial Institutions"
- "Currency Conversion"
- "WAY4™ Global Parameters"
- "Generating Reports in WAY4™"
- "Interest Accrual"
- "Issuing Module. User Manual"
- "Daily Procedures"
- "DB Manager User Management"

The following notation is used in the document:

- Field labels in screen forms are shown in *italics*.
- Button labels used in screen forms are shown in square brackets, such as [Approve].
- Menu selection sequences are shown with arrows, such as Configuration Setup →Contract Types.


Warnings about potentially hazardous situations or actions


Information about important features, additional options or the best use of certain system functions.

1 "SIC Group" Dictionary

The *SIC Group* parameter allows Services to be further defined according to the merchant category group.

1.1 Configuring "SIC Group"

To add a new SIC group, select the "Full →Configuration Setup →Main Tables → SIC Groups" user menu item.

As a result, the "SIC Group" grid form will appear (see Fig. 1).

SIC Groups << < > >> 1 of 14

Name	Group Code	Classifier Type
Cash	C	
Retail	+	
Airline	AI	VISA
Financial Institutions	Z	
Hospitalization/School Expense	O	
Hotel/Motel	H	
Mail/Telephone/Preauth order	T	
MasterCard MoneySend	MS	MC
Payment Services	P	
Restaurant	F	
Retail Purchase	R	
Transportation	X	

Ins Del Query Normalize SIC Codes Parents Sub Groups

Fig. 1. Grid for entering data on new SIC groups

In addition to fields for specifying a group name and code, the form contains the *Classifier Type* field. This is a drop-down list of group classifiers.

To create or delete a selected SIC group, use the [Ins] and [Del] buttons, respectively.

When registering a new SIC group, note the following:

- The *Group Code* field is mandatory.
- The combination of values set in the *Group Code* and *Classifier Type* fields must be unique in the "SIC Group" dictionary.

SIC groups can be included in other (higher) groups and form hierarchies.

To include a SIC group in a selected group, open a special form by clicking the [Sub Groups] button in the "SIC Groups" form.

To open the list of groups in which the selected group is included, click the [Parents] button.

Clicking on the [SIC Code] button in the "SIC Group" form will display a list of codes for the merchant categories belonging to this group.

To register merchant category codes, use the "SIC Codes" form (see Fig. 2), opened through the "Full →Configuration Setup →Main Tables →SIC Codes" user menu item.

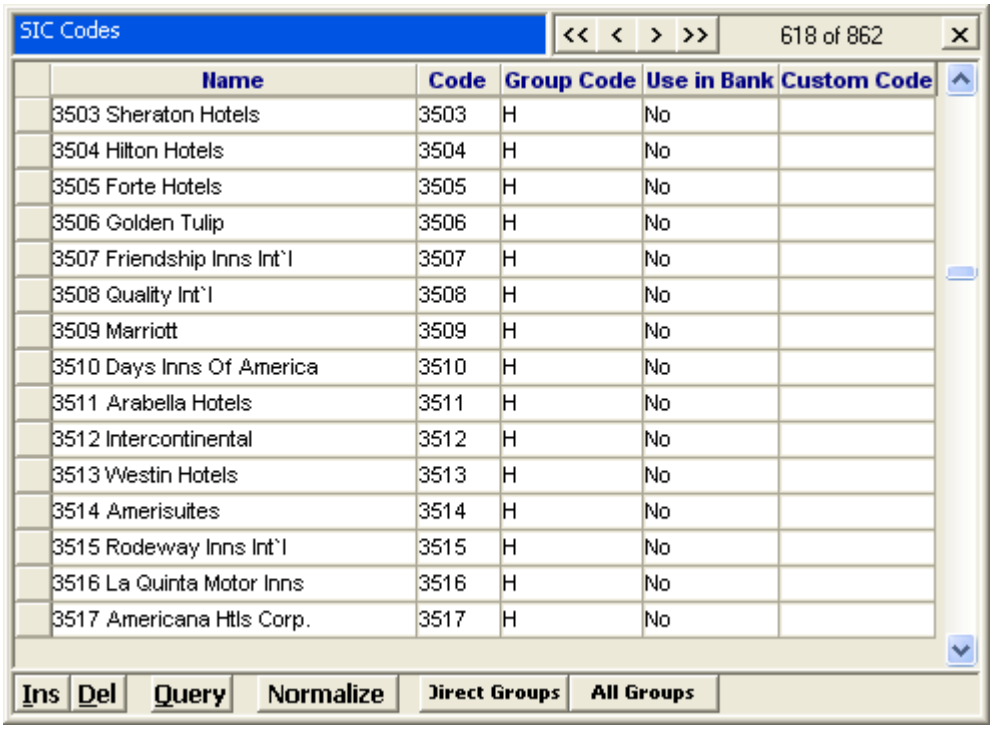


Fig. 2. Merchant category codes

To add a new SIC code or delete a selected SIC code, click the [Ins] or [Del] button, respectively.

When registering a new SIC code, note the following:

- The Code and *Group Code* fields are mandatory.
- The combination of values set in the *Code* and *Group Code* fields must be unique in the "SIC Codes" dictionary.

1.2 Including a SIC Code in a SIC Group

To specify the SIC group in which a merchant category code must be included:

- In the "SIC Codes" form, click the [Direct Groups] button for the selected code and open the "Direct Groups for <name of SIC code>" form.

- Click the [Ins] button and select the required group from the drop-down list.

Click the [Normalize] button in the "SIC Codes" or "SIC Groups" form after assigning a SIC group to a merchant category code so that the code is included in a certain SIC group or in all parent SIC groups. During the process for including the merchant code in a SIC group, field values and their uniqueness in the corresponding dictionary are checked:

- *Group Code* and *Classifier Type* ("SIC Group" dictionary).
- *Code* and *Group Code* ("SIC Codes" dictionary).

If errors are found, the message "Normalization has error: see process log" is generated.

Note that [Normalize] must be performed each time the composition of "child" groups changes. If a new SIC code is added to a "child" group, it will be added to all parent groups only after normalization. When the [Normalize] button is clicked, normalization will be performed immediately for all changed groups.

To open the list of groups in which the selected merchant category code is included, click the [All Groups] button in the "SIC Codes" form.

1.3 Using "SIC Group" to Search Services

The value of the *SIC Group* field (see the "Transaction Parameters" section of the WAY4™ Service Packages Administrator Manual) determines that this Service is only applicable for Service operations with retail outlets belonging to the indicated group.

A search for an appropriate Service is made according to the SIC group that includes the merchant. If normalization has been performed and a suitable Service was not found, all possible higher-ranking SIC groups in relation to this group will be analysed.

1.4 Using "SIC Group" when defining Usage Limiters

If normalization has been performed, limiters for the SIC group to which the merchant belongs will be considered, as well as those of higher-ranking groups that include the SIC group.. This allows, for example, for a separate limit to be placed on hotel transactions and a general limit for all retail operations.

2 Country Area Support

2.1 Support for Classifier "Area"

To support the use of areas in the system, a special classifier "Area" has been created. It is used to route transactions, search Services and specify forbidden retail organisations.

Areas are defined in the Country Area dictionary ("Full →Configuration Setup → Main Tables →Country Area"), which is set up in the same way as the SIC Group dictionary (see the section ""SIC Group" Dictionary"). One area can be a part of other areas.

Every country may belong to any area. The area to which a country belongs is determined by the value of the *Area* field in the Country Table dictionary ("Full → Configuration Setup →Main Tables →Country Table").

Country classifications are set by the system administrator.

For more flexible definition of an area, use the political divisions dictionary ("Full →Configuration Setup →Main Tables →Country Objects"), in which states and cities belonging to countries in the "Country Table" dictionary are specified.

2.2 Configuring Country Areas

To view the list of countries in an area, use the "Full →Configuration Setup →Main Tables →Country Area" user menu item.

The screen will display the "Country Area" grid (see Fig. 3).

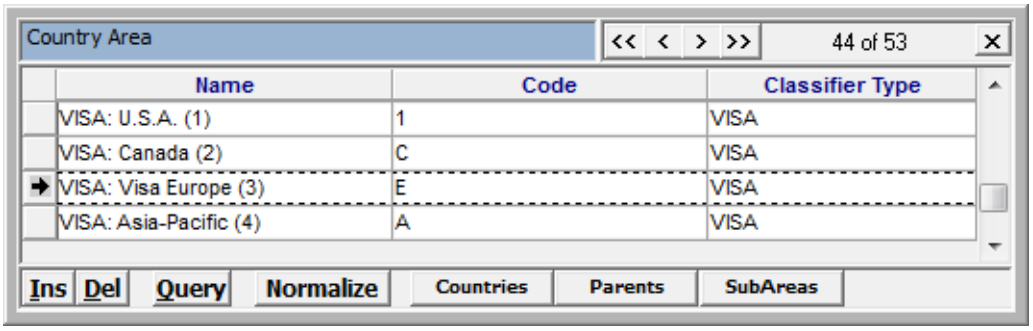


Fig. 3. Grid for configuring areas

In addition to fields for specifying an area name and code, the form contains the *Classifier Type* field. This is a drop-down list of area classifiers used to group areas.

To add a new record or to delete a selected area, use the [Ins] and [Del] buttons, respectively.

Areas can be included in other (higher) areas and form hierarchies.

To include an area in a selected area, open a special form by clicking the [SubAreas] button in the "Country Area" form.

To open the list of areas in which the selected area is included, click the [Parents] button.

To access the list of counties belonging to the selected area click on the [Country] button.

2.3 Registering Countries

Countries are registered in the "Country Table" form (see Fig. 4), opened through the "Full →Configuration Setup →Main Tables →Country Table" user menu item.

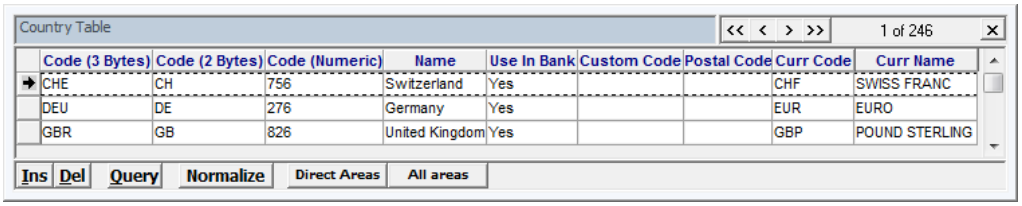


Fig. 4. List of countries

To add a new country or delete a selected country, click the [Ins] or [Del] button, respectively.

The *Curr Code* and *Curr Name* fields are used in "Dynamic Currency Conversion" functionality to define card currency.

To specify the area in which a country must be included, click the [Direct Areas] button in the "Country Table" form and add the necessary area in the "Direct Areas for <name of country>" form by clicking the [Ins] button and selecting the area from a drop-down list.

Note that when searching for a Service (see the section "Using Preferred Counterparties to Search Services") the system checks that the area specified in preferred counterparty parameters (if specified) includes the country of the merchant where the transaction was performed. This means that the system only analyses the area specified in the preferred counterparty parameters and all its parent groups. The preferred counterparty parameters may contain a parent area of the area in which the country is included in the "Direct Areas for ..." form. Click the [Normalize] button in form "Country Table" or "Country Area" after

assigning an area to a country so that the country is considered included in all parent areas.

To open the list of areas in which the selected country is included, click the [All areas] button in the "Country Table" form.

2.4 Configuring Political Divisions

To show the list of political divisions (states, cities) specified in the system, select the user menu item "Full → Configuration Setup → Main Tables → Country Objects".

The "Country Objects" form will open (see Fig. 5).

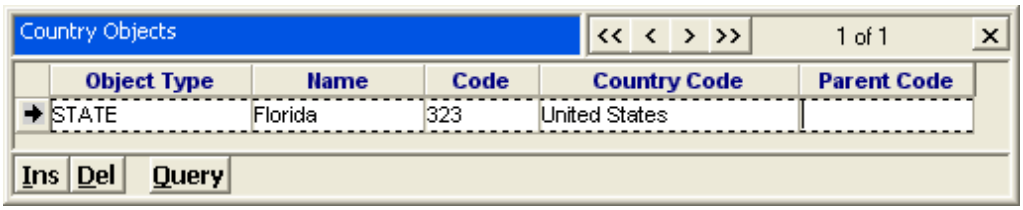


Fig. 5. Form for configuring political divisions

To add a new record or delete a selected record, click the [Ins] or [Del] button, respectively.

When adding a new record, the following fields must be filled in:

- *Object Type* – the type of political division (object type); selected from a list ("STATE" or "CITY")
- *Name*- the object name
- *Code* – the object code
- Country Code – the name of the country to which the object belongs.
- Parent Code – the code of the parent object (for example, for a state – the country code; for a city – the state code)

2.5 Using Areas in WAY4

Areas are used when processing authorisations or financial documents, as well as in the following situations.

2.5.1 Searching Routing Contracts in the Interchange Routing Table

During routing contract search, it is possible, for instance, to set special conditions for transactions from "neighbouring" or "dubious" countries.

When searching for a card's routing contract, the area is determined by the country code in the BIN table. When searching for a device's routing contract, the

area is determined by the country code indicated in the document. If a routing contract does not specify an area, it is valid for all areas.

2.5.2 Using Preferred Counterparties to Search Services

The preferred couterparties list contains the *Area* field, which allows for filtering operations by area.

For cards, the area is determined through the country code in the BIN table. For devices, the area is determined through the country code indicated in the document.

2.5.3 Analysing the List of Forbidden Retail Outlets

The list of forbidden retail outlets (see the "Configuring Preferred Counterparties" section in the Preferred Counterparties Administrator Manual) contains the *Area* field, which allows operations to be filtered by area.

The area is determined by the country code indicated in the document.

3 Business Calendar

The business calendar automatically determines which days are business days and which days are weekends and non-working days. The system checks the business calendar setup when a new banking day is started (see the "Start of Day Procedure" section in the Daily Procedures User Manual), when the acquiring bank of a retail organisation determines the due date (see the "Posting" section in the Service Packages Administrator Manual) and in some other situations.


It should be noted that the number assignment of days in the system is fixed according to the European numbering convention: Monday is "1", and Sunday is "7".

Different financial institutions registered in WAY4 may use different business calendars. For this, it is possible to register different business calendar types in the system (Full → Configuration Setup → Main Tables → Calendar Types). The used business calendar type is specified in financial institution parameters (the *Calendar Type* field of the "Details for <name of FI>" form). To use different business calendars, enable Time Zones mode, see the document "Time Zones".


A separate agreement with the WAY4 vendor is required to use Time Zone mode.

In WAY4 it is possible to use business calendars that differ from the default calendar (financial institution's calendar), for example, to calculate functional dates, activate payment orders, open/close Events, charge nontransaction fees, to determine the period funds are blocked when processing an authorisation request, etc. To do so, the CALENDAR_TYPE=<name of business calendar type> tag is used. This tag is set for the functional date, payment order, Event, transaction subtypes, or Service.

Due normalization macrotransactions can also be generated and posted according to a separate business calendar that differs from the financial institution's business calendar:

- To post a due normalization for a specific account, a contract functional date with the CALENDAR_TYPE=<name of business calendar type> tag can be used.
- The CALENDAR_TYPE=<name of business calendar type> tag can be set in an account template's *Template Details* field.

The specified calendar type will be used when calculating payment dates for accounts created according to this template.

- The CALENDAR_TYPE=<name of business calendar type> tag can be set on the Account Scheme level. In this case, the specified calendar type will be used for all templates in this Scheme.

  For example

  September 6 is specified in the financial institution's business calendar as a business day. If for any reason it is necessary to move the processing of due normalization transactions from September 6 to September 7, in the account template, specify a business calendar in which September 6th is not a business day. The corresponding due normalization transactions will be generated and processed on September 7. Other types of transactions will be processed according to the financial institution's business calendar.

  Business calendars are created manually or are imported to WAY4 by a pipe.

3.1 Manually creating a business calendar

To fill in the business calendar, select the "Full →Configuration Setup →Main Tables →Business Calendar" user menu item. As a result, the screen will display the "Business Calendar" grid (see Fig. 6).

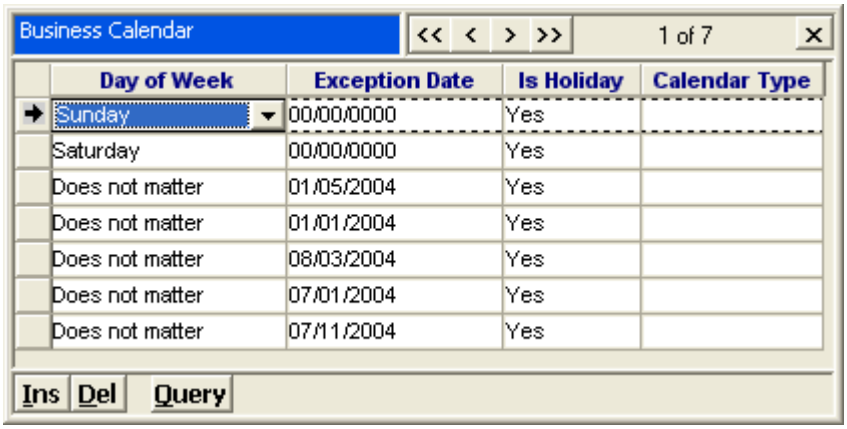


Fig. 6. Business calendar grid

The "Business Calendar" grid contains the following fields:

- *Day of Week* – day of the week
- *Exception Date* – field for indicating the date in the DD/MM/YYYY format
- *Is Holiday* – specifies whether or not this day is a working day ("Yes"/"No")

- *Calendar Type* – calendar type selected from a list. When this field is filled in, the corresponding date belongs to the selected business calendar.

  If the field is not filled in, the corresponding setting for this date (or day of the week) belongs to the default business calendar. The default business calendar is used if a specific calendar type is not defined for the financial institution (or in the account template). If a specific calendar type is specified for the financial institution (or in the account template), this default calendar setting is ignored.

  It's important to note that if a date in a calendar is not defined as either a non-working or working day, by default it will be considered as a working day in this type of calendar (with the exception of cases when this day is a Saturday or Sunday, which are marked in this calendar as non-working days regardless of the date.

  For example, in a "default" calendar (with an empty Calendar Type field), 1 January 2016 (Friday) is a holiday. In another calendar with the Calendar Type field filled in (for example, with the value "CALENDAR_2"), 1 January 2016 is not defined in the calendar (i.e. this day is not defined either as a non-working or working day and Fridays are not a non-working days). For financial institution 003, Calendar Type = "CALENDAR_2" is specified. In this case, in all processes in financial institution 003, 1 January will be a working day.

To create a new record, click on the [Ins] button and fill in the fields according to the following rules.

To configure the system to recognise a weekend day (such as Sunday), set the day in the *Day of Week* field, leave the *Exception Date* field set to 00/00/0000, and set the *Is Holiday* field to "Yes".

To configure the system to recognise a specific date as a non-working day (such as 1 January 2009), the grid should be filled in as follows. The *Day of Week field* should have the "Does not matter" value, the *Exception Date* field, the date 01/01/2009, and the *Is Holiday* field should contain the "Yes" value.

A weekend day that is a working day is set up in the same way (for example, Saturday, 3 January 2009 is a working day). In this case, the fields are filled in as follows: the *Day of Week* field should contain the value "Does not matter", the *Exception Date* field, 03/01/2009, and the *Is Holiday* field should read "No".

The *Day of Week* field cannot specify a certain day of the week when the *Exception Date* field indicates a certain day (for example, "Monday", "12/01/2009").

3.2 Using a pipe to import a business calendar

The Java pipe com.openwaygroup.pipe.business_calendar_import.jar is used to import a business calendar.

To run the Java pipe, select the user menu item "Full →Configuration Setup → Main Tables →Business Calendar Import".

Business Calendar Import pipe parameters

Parameter	Default value	Req.	Parameter description
SOURCE_DIR	@INTERCHANGE_PATH@\calendar\in	M	Incoming file directory into which the imported file should be put.
PROCESSED_DIR	@INTERCHANGE_PATH@\calendar\arch	M	Directory of files that have been successfully processed.
ERROR_DIR	@INTERCHANGE_PATH@\calendar\err	M	Directory with files whose import failed.
CALENDAR_TYPE		O	Parameter in which the business calendar type can be specified. For example, "Holidays".

The following letters indicate whether or not it is mandatory to fill in a field:

- M – the field must be filled in
- O – the field is optional

An imported file in CSV format is put into the SOURCE_DIR directory. This file's header corresponds to the mask "CALENDAR_*.csv".

The pipe makes it possible to import a business calendar with an *Is Holiday* attribute of "Yes".

When importing a business calendar, the following checks are made automatically:

- Imported dates (*Exception Date*) are set and their value is greater than the current banking date.
- There is no calendar with this type (*Calendar Type*) in the CALENDAR table.

3.2.1 Type of imported business calendar

If the CALENDAR_TYPE parameter is not set, the calendar type is determined from the file name: the second element after the static identifier "CALENDAR" will be used as the calendar type. For example, if the file name is "CALENDAR_IamCalendarType_2020_01.csv", the string "IamCalendarType" will be considered to be the *Calendar Type* value. If import of calendars with different *Calendar Type* values is planned, it is recommended to determine the calendar type from the file name. If the pipe always imports the same calendar

type, it is recommended to set this type in the CALENDAR_TYPE parameter. The value of the CALENDAR_TYPE pipe parameter has a higher priority than the value that is set in the file name. In addition, the pipe supports the ability to import business calendar dates without specifying *Calendar Type.* In this case, the file must have the name "CALENDAR_DEFAULT_2020_01.csv".

3.2.2 Examples of an imported CSV file

The first line in the file lists entities being imported, and the values of these entities are specified in each subsequent line:

```
Date,Name,Year,Month,Day,Weekday,Types,IsObserved,Country,Locations,States
2021-01-01,"New Year's Day",2011,1,1,Sat,National holiday,,United States,,
2021-01-17,Martin Luther King Day,2011,1,17,Mon,National holiday,,United States,,
2021-02-14,"Valentine's Day",2011,2,14,Mon,Observance,,United States,,
2021-02-21,"Presidents' Day",2011,2,21,Mon,National holiday,,United States,,
```

The "Name" and "Country" values are not processed and are reserved for forward compatibility.

Only *Exception Date* values can be imported by the pipe, and in this case, the imported file will appear as follows:

```
Date
2021-01-01
2021-01-17
2021-02-14
2021-02-21
```

4 "Message Channels" Dictionary

The Message Channels dictionary is used to define identifiers of logical routes (channels) in WAY4. These identifiers serve to select rules for processing transaction information.

Note that changes to this dictionary can only be made under supervision of the WAY4 system vendor representatives.

To access the dictionary, use the "Message Channels" form (see Fig. 7). It is opened by selecting the "Full →Configuration Setup →Main Tables →Message Channels" menu item.

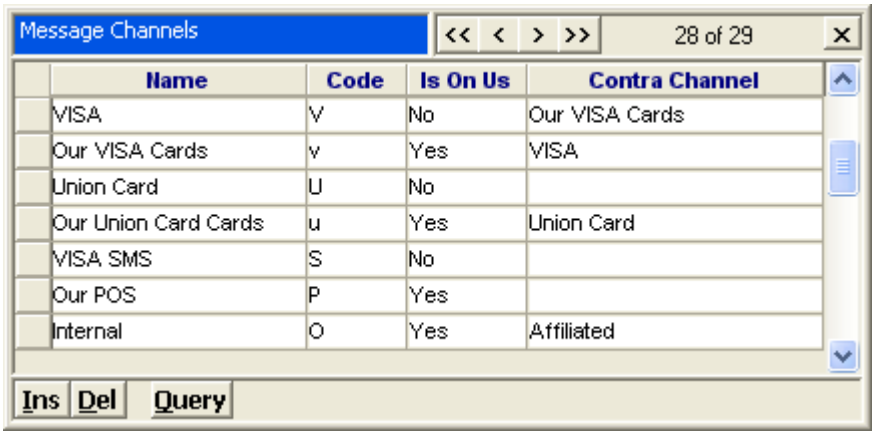


Fig. 7. "Message Channels" dictionary

The form contains the following fields:

- *Name* – name of the logical route (channel)
- *Code* – code of the logical route (channel)
- *Is On Us* – one of the following values can be selected in this field:
  - "Yes" – the channel is used to interact with an internal subsystem of the WAY4 system
  - "No" – the channel is used to interact with an external system
  - "Affiliated" – the channel is used to interact with an external system installed in a bank that has additional agreements with our bank. This channel type can be used to implement interaction between a sponsor bank and an affiliated bank.
- Contra Channel – this field's value is used as the transaction counterpart's channel if it is impossible to determine the channel through standard routing mechanisms.

5 Message Dictionary

The "Message Dictionary" contains a list of WAY4™ system objects with text equivalents of their names or text messages generated by the systemwhen working with these objects. The dictionary's names, text messages and descriptions of them can be translated into local languages.

5.1 General Information

The "Message Dictionary" grid form (Full →Configuration Setup →Main Tables → Message Dictionary (see Fig. 8)) is used by various WAY4 modules and contains a significant number of records. Therefore, to open the dictionary it is recommended to use preliminary data retreival (see the section "Preliminary Selection of Records according to Arbitrary Criteria" of the "DB Manager Administrator Manual"). For example, to open the list of transaction subtype codes, (see Fig. 8) use the following selection criteria: *Object Type* = "Transaction" and *Object Name* = "GL Transaction".

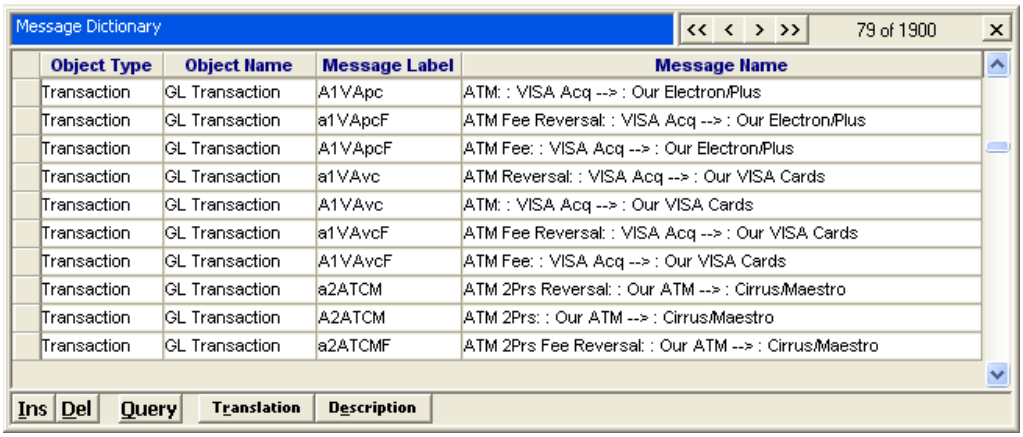


Fig. 8. "Message Dictionary" system dictionary

The value of the record code in the "Message Dictionary" is contained in the *Message Label* field, and its corresponding name (see Fig. 8) or message text is contained in the *Message Name* field.

WAY4 makes it possible to translate names or messages contained in the *Message Name* field into local languages registered in the corresponding system dictionary ("Full →Configuration Setup →Client Classifiers →Languages"), "Full → Configuration Setup →Languages and Localisation →Languages", see the section ""Languages" dictionary"). For more information, see the section "Translation of names, messages and descriptions".

The [Description] button is used to access a detailed description of the message contained in the *Message Name* field.

5.2 Translation of names, messages and descriptions

To translate the name, message text or the description of a message contained in the *Message Name* field into a local language, select the required row in the "Message Dictionary" grid form and click [Translation].

The form "Translation for <name>" will appear on the screen (see Fig. 9).

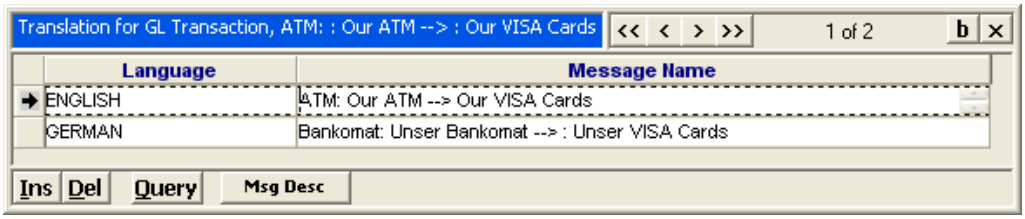


Fig. 9. Grid form for translation into local languages

To add a translation of a name or message text, click the [Ins] button to add a row to the grid form, select the required language from the drop-down list in the *Language* field and in the *Message Name* field, enter the required text in the selected language.

To add the translation of a message description (detailed description of a message) contained in the *Message Name* field, click the [Msg Desc] button and add the translation in the *Description* field of the form that opens.

Note that in most cases the contents of the *Message Name* field consist of separate segments, for example, (see Fig. 9): <Transaction>::<Source Contract Type>-->:<Target Contract Type> or <Account Type><Service Class>, etc. Moreover, separate segments of messages, for example transaction type (Transaction) are also elements of the "Message Dictionary". Therefore, for messages with the parameter Object Type = "Transaction" it is possible to translate into a local language an entire separate message corresponding to a particular transaction, as well as a message segment as an element of the "Message Dictionary". If a separate segment of a message is translated into a local language, after the dictionary is refreshed, this translation will be used by the system for all messages with the parameter Object Type = "Transaction" containing the given segment.

Message segments that are separate elements of the "Message Dictionary" include:

- Transaction type (Object Name = "Transaction")
- Contract type (Object Name = "Contract Type")
- Values of transaction type classifiers (Object Name = "Service Class")
- Account types (Object Name = "Account Type")
- Elements with the Object Name parameter with the "Fee", "Fee Reversal", Reversal", Information", "Information Advice" values.

To search the "Message Dictionary" dictionary for separate message segments, use preliminary data retreival (see the section "Entering and Editing Data" of the "DB Manager Administrator Manual") with the following criteria: Object Type = "Transaction", Object Name = "Transaction"/ "Contract Type"/ "Service Class"/ "Account Type"/ "Fee"/ "Fee Reversal"/ "Reversal"/ "Information"/ "Information Advice".

5.3 Updating the "Message Dictionary"


Note that registration of new transaction subtypes and new account types, as well as new translations of message names and texts (Message Name) into local languages requires the "Message Dictionary" system dictionary to be updated.

When the "Message Dictionary" is updated, transaction subtype uniqueness is checked in TRANS_TYPE table (*TransType Idt* and *RBS Code*) and if duplicates are found:

- The *TransType Idt* field's value is automatically corrected.

  If *TransType Idt* is not filled in, it will be filled in automatically when the check is performed.

- For the *RBS Code* field, when a duplicate is found, the record is not updated and an error is logged to the process log ("Full →Process Log → Process Log").


Note that if changes were made in transaction subtype names, account type names, etc., the *Message Name* field in the MESS_DICT table is not automatically updated. For data consistency, either manually edit the *Message Name* field value in the dictionary, or delete the record with the obsolete name from the dictionary and start the update process. When deleting records from the dictionary, it is important to note that these records, for example, Contract Type, may be included in other records from the dictionary, for example, in entry

descriptions according to transaction types. In this case, a number of entries may need to be deleted before starting the update process.

The "Message Dictionary" is updated using the menu item "Full →Configuration Setup →Main Tables →Generate Message Dictionary".

During execution of this menu item, the screen will display the dialog window "Get Language Trans" (see Fig. 10) used to select one of the languages registered in the system for which the dictionary should be updated.

When updating the "Message Dictionary" dictionary, specify an empty value, which corresponds to the system's main (English) language, in the *Language* field of the "Get Language Trans" dialog window (see Fig. 10); click the [Proceed] button and confirm execution of the update by clicking the [Yes] button in the "Confirm Execution" window (see Fig. 11). As a result, the dictionary will be refreshed in the system's main language.

Fig. 10. Dialog window for selecting the language in refreshing the "Message Dictionary"

To refresh the "Message Dictionary" in a local language, select the required language in the *Language* field of the "Get Language Trans" dialog window (see Fig. 10).

After selecting the required language, using the "Yes" or "No" values, specify in the *Override translations* field whether existing translations of names (Message Name) will be updated in the specified language, click the [Proceed] button and confirm execution of the update by clicking the [Yes] button in the "Confirm Execution" window (see Fig. 11).

After updating the "Message Dictionary" dictionary in the local language, an entry for this language will be created for every element of the dictionary in the "Translation for <name>" grid form (see Fig. 9 of the section "Translation of

names, messages and descriptions"), with the *Message Name* field value in the main language if no translation for message segments exists. If a translation into a local language exists for message segments and the "Yes" value is specified in the the *Override translations* field of the "Get Language Trans" dialog window (see Fig. 10), the corresponding translations will be used by the system when refreshing all records in that language. When the value of the *Override translations* field of the "Get Language Trans" dialog window is "No", records will be created in the local language (using existing translations of message segments) only for those elements of the "Message Dictionary" for which such records were absent.

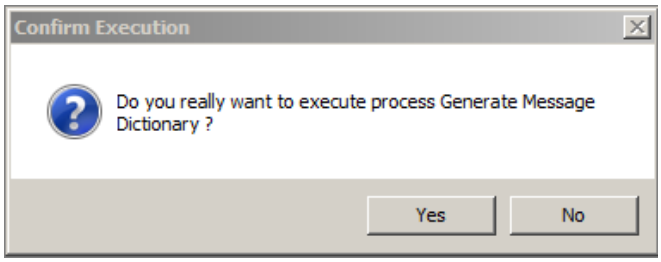


Fig. 11. Confirmation request for update of "Message Dictionary" dictionary

Note that the system stores records with translations into local languages in the same database as "Message Dictionary" records in the system's main language (English). This means that the form editor (see the "Form Editor" document) can be used with this form to create a special form that will show only those records containing a translation into the corresponding language.

5.4 Excluding Duplicate Records from the "Message Dictionary"

The "Message Dictionary - Duplicates" form, menu item "Full →Configuration Setup →Main Tables →Message Dictionary - Duplicates" is used to search for duplicate records with the same values in the *Object Type*, *Object Name*, *Message Label*, and *Language* fields of the "Message Dictionary".

If there are duplicates in the "Message Dictionary", they are shown in the "Message Dictionary - Duplicates" form (see Fig. 12).

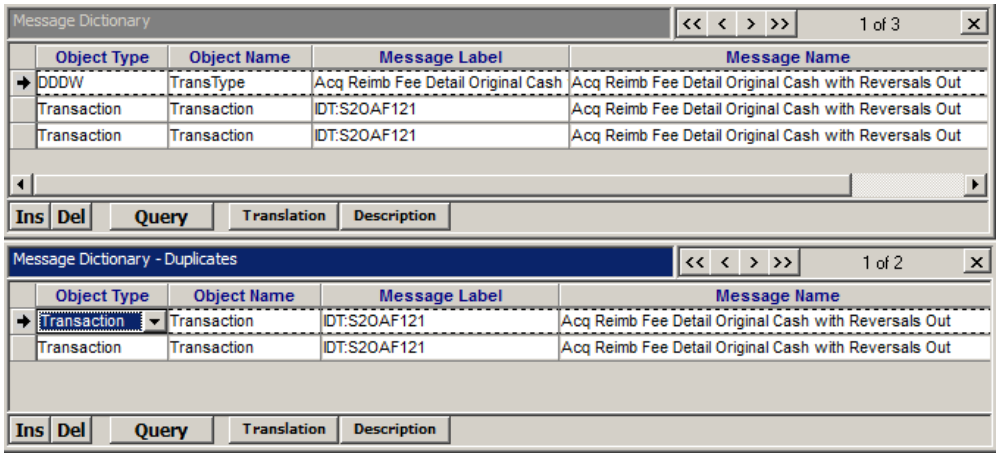


Fig. 12. Duplicate records in the "Message Dictionary"

To avoid incorrect system behaviour, it is recommended to delete duplicate records. The activity is performed with the [Del] button in the "Message Dictionary - Duplicates" form.

5.5 Deleting incorrect GL Transaction records

If a transaction type for which records were already created in the "Message Dictionary" changes (for example when the transaction type's *RBS Code* or *RBS Rev Code* fields change) or is deleted, delete the corresponding records in the "Message Dictionary" so that the transaction will be shown correctly in WAY4. One of the following menu items can be used to do so:

- "Full → Configuration Setup → Main Tables → GL Transaction Message Dictionary Synchronize"
- "Full → Configuration Setup → Transaction Types → GL Transaction Message Dictionary Synchronize"

These menu items open the "Confirm - Delete" form. Select a record in the form's *Delete Records* field (see Fig. 13):

- "No" – records are only check but incorrect records are not deleted. The check's results can be viewed in the process log.
- "Yes" – records are checked, and incorrect records are deleted. The results of the process can be viewed in the process log.

  During the check, a search is made for the corresponding record in the list of transaction types (by the *RBS Code* or *RBSRev Code* field). If no record in the list of transaction types was found for the record from the "Message Dictionary" and "deletion mode" is used, the record in the "Message Dictionary" gets the value "C" in the AMND_STATE field and the "deletion" date is specified in the AMND_DATE field.

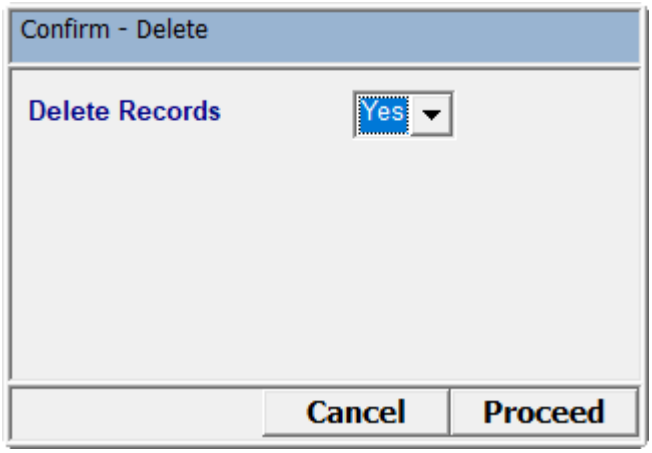


Fig. 13. "Confirm - Delete" form

6 Service Groups Dictionary

Service Groups is an additional client and contract classifier. For example, according to additional classification, VIP clients can be distinguished as a separate group.

Service Groups can be used when configuring various filters for viewing data or generating reports. In particular, a Service Group can be set on the local constants level for filtering data available to users when working in DB Manager forms ("Full →DB Administrator Utilities →Users & Grants →User Groups and Users - View →[Constants]"), for more information see the section "Initialising Local Constants" of the document "DB Manager User Management".

Service Groups are configured in the "Service Groups" form ("Full → DB Administrator Utilities →Users & Grants →Service Groups"), see Fig. 14.

Clients and contracts are marked using configured Service Groups, for example, when registering clients and contracts with applications (see the document "Advanced Applications R2").

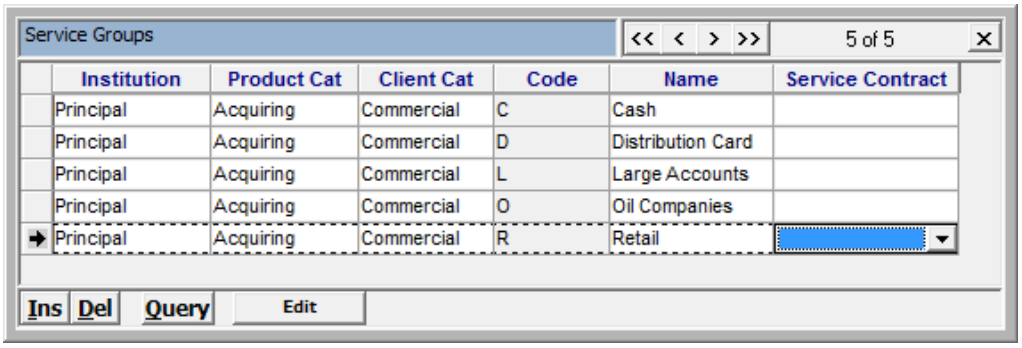


Fig. 14. Service Groups dictionary

To create a new Service Group, click the [Ins] button and fill in the fields of the new record.

The form contains the following fields:

- *Institution* – financial institution for which this Service Group will be available.
- *Product Cat* – Product category:
  - Issuing – Products for issuing contracts.
  - Acquiring – Products for acquiring contracts.
  - Accounting – Products for bank system contracts. This value has been kept for backward compatibility.
  - Bank Accounting – Products for bank contracts.

- *Client Cat* – client type ("Private" – individual, "Commercial" – legal entity, "Accountant" – bank department).
- *Code* – Service Group code. After filling in the *Code* field, this field becomes unavailable for editing in the "Service Groups" form. See the note below.
- *Name* – name of Service Group.
- *Service Contract* – Service contract. A Service contract is selected from a list of contracts registered in WAY4 and can be used, for example, to receive notifications for ATM status messages.

After entering the Service Group code, the *Code* field closes for editing in the "Service Groups" form in order to avoid accidental changes in the value of this field. A client or contract is linked with the Service Group assigned to it through the Service Group code (the group code is entered in the *Service Group* field of the client or contract form). If the group code is changed in the the "Service Groups" form after the Service Group is assigned, links to this Service Group from the properties of the corresponding clients and contracts will be violated.

If it is necessary to change the Service Group code, do as follows:

- Click the [Edit] button in the "Service Groups" form. The "Edit for <name of Service Group>" form will open, in which the *Code* field will be editable (see Fig. 15).
- If changes are made to a Service Group code that is already assigned to clients and contracts, the Service Group code must be manually synchronised in the "Service Groups" form and in the properties of the corresponding clients and contracts.

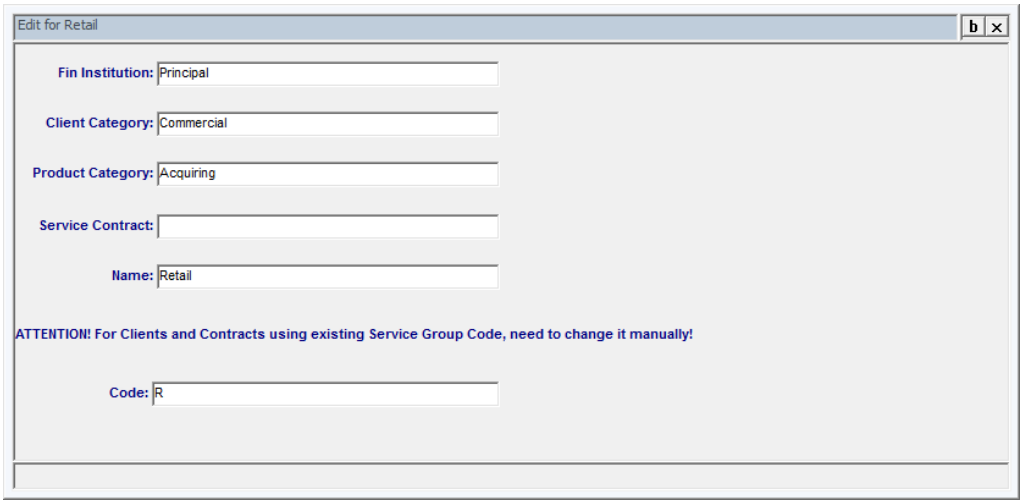


Fig. 15. "Edit for <name of Service Group" form

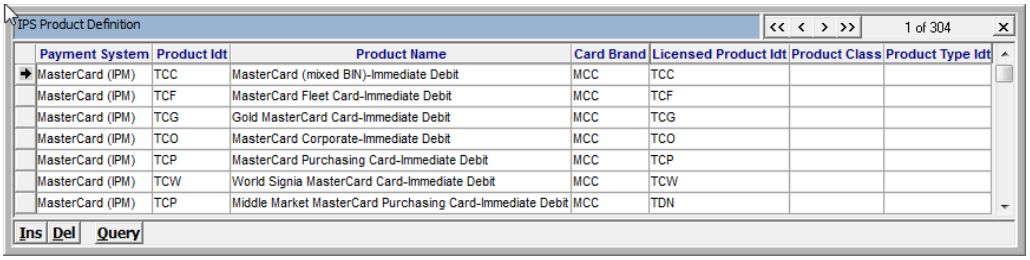
In the "Edit for <name of Service Group>" form, a warning is displayed that is is necessary to synchronise the changed Service Group code and client and contract properties:

"ATTENTION! For Clients and Contracts using existing Service group Code, need to update it manually!".

7 Payment System Product Dictionary

This dictionary is used to manage public names of payment system card products and a number of their parameters used, for example, when generating payment system statistic reports.

The dictionary is opened with the user menu item "Full → Configuration Setup → Routing →IPS Product Definition" (see Fig. 16).


IPS Product Definition

<< < > >> 1 of 304

Payment System	Product Idt	Product Name	Card Brand	Licensed Product Idt	Product Class	Product Type Idt
MasterCard (IPM)	TCC	MasterCard (mixed BIN)-Immediate Debit	MCC	TCC		
MasterCard (IPM)	TCF	MasterCard Fleet Card-Immediate Debit	MCC	TCF		
MasterCard (IPM)	TCG	Gold MasterCard Card-Immediate Debit	MCC	TCG		
MasterCard (IPM)	TCO	MasterCard Corporate-Immediate Debit	MCC	TCO		
MasterCard (IPM)	TCP	MasterCard Purchasing Card-Immediate Debit	MCC	TCP		
MasterCard (IPM)	TCW	World Signia MasterCard Card-Immediate Debit	MCC	TCW		
MasterCard (IPM)	TCP	Middle Market MasterCard Purchasing Card-Immediate Debit	MCC	TDN		

Ins Del Query

Fig. 16. "IPS Product Definition" list

This form contains the following fields:

- *Payment System* – payment system name.
- Product Idt:
  - For MasterCard – the identifier defining the payment system product in the MasterCard clearing centre (GCMS (Global Clearing Management System) Product Identifier).

    For MasterCard, there may be several records with the same value in the *Product Idt* field. These records differ by the value in the *Licensed Product Idt* field (the value of the *Product Name* field for these records may be the same or differ).
  - For VISA – the identifier of the product in VISA.
- *Product Name* – name of the card product in the payment system.
- *Card Brand* – three-character identifier indicating the payment system product programme, for example, "DMC" – MasterCard Debit. For MasterCard, the value of this field corresponds to the *Brand* field of the "BIN Table" form. For Visa, the value "VISA" or "PLUS" is specified in this field.
- *Licensed Product Idt* – identifier of the product in MasterCard. For other payment systems, the value in this field duplicates the value of the *Product Idt* field.
- *Product Class* – field reserved for future use.

- *Product Type Idt* – field reserved for future use.

The bank's own products are marked by a generalised product with the value "Internal" in the *Payment Scheme* field and the value "Private Label" in the *Name* field.

Public names of card products and their parameters are defined in the following cases:

- When card contract subtypes are marked with payment system products. Marking may be performed as follows:
  - When the menu item "Contract Types Validation" ("Full →Configuration Setup →Contract Types →Contract Types Validation") is executed. All "on-us" card product subtypes are marked (for example, "one-time" marking after data are loaded into the BIN table). Only active subtypes are checked (with the "Yes" value in the *Is Active* field).
  - When the [Validate] button is clicked in the card contract type form ("Full →Configuration Setup →Contract Types →Card Contract Types"). In this case, all subordinate subtypes are marked.
  - When the [Validate] button is clicked in the contract subtype form ("Full →Configuration Setup →Contract Types →Card Contract Types → [Sub Types]"). In this case, a specific contract subtype is marked (for example, when a new subtype is added).

  A search is made for a record in the BIN table corresponding to this sub-type. The search is made by the *Sub BIN* field and the card range (*Start BIN*, *End BIN* fields).
  - In the contract sub-type's *BIN Record* field, the values of the following fields from the "BIN Table" form are specified in the following format: <*Sub Bin* field value>:<*Product ID* field value, i.e. IPS product name>:<*Usage* field value>.
  - The *IPS Product* field of the contract subtype is filled in with the value of the *Product Name* field from the "IPS Product Definition" form.

For MasterCard, this field is filled in with the value of the *Product Name* field of the record for which the values of the *Licensed Product Idt* field in the IPS_PRODUCT table and *Licensed Product Idt* field in the BIM_TABLE table correspond.

See the document "Products and Contract Subtypes".

- When processing (Acceptance) a document to fill in the the BIN_RECORD field in the DOC table (or *Card Bin* field in the "Primary Doc – Full" form):
  - When processing a document for on-us cards, the contract sub-type's *BIN Record* field value is specified in this field.
  - When processing a document for foreign cards, the reference to the BIN table row according to routing settings is specified in this field (the identifier of the record in the BIN table is specified).

8 Contract Statuses

Contract statuses are registered in the "Contract Statuses" form ("Full → Configuration Setup →Contract Types →Contract Statuses"), see Fig. 17.

Contract Statuses — 23 of 42

Category	Name	Code	Is Valid	External Code	Code Parms	Chip Card Action Type
Card	My Status for Card	21	Valid	21		
Card	Honour with identification	08	Valid	08		
Card	Credit loss	05	Decline	T60		
Card	Card OK	00	Valid	00		
Card	Card No Renewal	50	Valid	50	DOC_RC=0;	
Card	Card Do not honor	05	Decline	05		
Card	Card Closed	14	Invalid	S40		
Card	Call Issuer	01	Decline	01		
Account	OVD Bucket 4	05	Decline	T40		
Account	OVD Bucket 3	05	Decline	T30		
Account	OVD Bucket 2	05	Decline	T20		
Account	OVD Bucket 1	00	Valid	T10		
Account	Non performing invalid	05	Invalid	T50		
Account	Non performing	05	Decline	T50		
Account	Migration temporary	MG	Invalid	MG		
Account	Credit loss	05	Decline	T60		
Account	Card Closed	14	Invalid	14		
Account	Account OK	00	Valid	00		
Account	Account Decline	05	Decline	05		
Account	Account Closed	14	Invalid	14		

Ins Del Query

Fig. 17. List of contract statuses

This form contains the following fields:

- *Category* – specifies the contract categories for which this status can be used (Card, Account, Device).
- *Name* – status name.
- *Code* – code of the status in WAY4. Used during authorisation (determines the response code that will be sent in response to an authorisation request).
- *Is Valid* – this field can have the following values:
  - "Valid" – authorisations, financial document posting and card issuing is permitted.
  - "Decline" – authorisations are prohibited, financial document posting and card issuing are permitted.
  - "Invalid" –authorisations, financial document posting and card issuing are prohibited. However, operations like interest accrual, due normalisation, etc. are performed.

Note that the "Invalid" status set in the top contract does not apply to the entire contract tree. When posting a financial document, only the status of the contract for which this document is being posted is considered. The status of the top contract is not considered.

This indicator can be used, for example, to set rules for changing a contract status according to an Event (if the FROM_VALID tag is set in the Event type, the contract will be transferred to the status set in the *New Status* field of the Event type only for a contract in a status with the "Valid" indicator).

- *External Code* – contract status code received from an external system. For example, when a status changes according to an imported application. The *External Code* code must be unique in the specific contract category.
- *Code Parms* – additional parameters are specified in this field as tags:
  - PR=<priority value> – specifies the priority of the status. The priority value is set in numeric form (0, 1, etc.) and a higher value indicates a higher priority. This parameter is used, for example, when changing a contract status using an Event (see the description of the NOT_BETTER tag in section "Tags used when working with Events" the document "Setup Tags").

    If the PR tag is not set for the status:

    - If the PR tag is not set, and the *Is Valid* field contains the "Valid" value, the status priority is equivalent to "0".
    - If the *Is Valid* field contains any other value, the status priority is equivalent to "1".
  - CLOSE_ACNT – when a status with this parameter is specified, the contract closes (the closing date of the contract will be equal to the current banking date). After the contract has been closed ("Is_Ready=Closed") no operations are performed with this contract's account.
  - CARD_MRK_EN=Y – by default, it is not permitted in WAY4 to mark cards for reissue in any status other than "Card Ok". The CARD_MRK_EN parameter with the "Y" value set in contract status properties allows a card with this status ("Card Ok") to be marked. For example, it may be necessary to reissue a card that has been temporarily locked (when the card status is "Call Issuer").
  - The DOC_RC=<response code>; tag specifies the response code sent when posting documents for a contract. Usually, the DOC_RC=0; tag is

used. A tag with this value is used so that contract status does not affect document posting when card status is "Card No Renewal" (so that in this status a financial document for a contract is not rejected).

- *Chip Card Action Type* – action that must be executed with a Smart card when a contract is assigned the corresponding status. When processing an online request for a card with this status, a script for executing the specified action will be generated and sent to the card.

9 "System Instances" Dictionary

The "System Instances" dictionary is used to register systems belonging to the bank's general infrastructure, in particular:

- WAY4 test system(s)

  Systems defined as test systems for the following modules "WAY4™ Product Inspector" (see the document "WAY4™ Product Inspector Module"), "WAY4 Real-Time Risk Management" module, recalculating usage limiters (see the document "Risk Monitoring"), "WAY4 Datamart (see the document "Configuring WAY4 for Transferring Data to Datamart") these modules are supplied according to separate agreements with the WAY4 vendor.

  Test systems for these modules are registered in the simplified form "System Instances - Simple" (Full →Configuration Setup →Main Tables → System Instances – Simple).

  If a system is registered as a test system in the "System Instances – Simple" list, a number of procedures and scenarios can be executed for it that cannot be executed in a production system. For example, recalculation of usage limiters when testing limiter settings can only be executed in a test system (see the document "Risk Monitoring").

  If a system is not registered in the "System Instances – Simple" list, it is considered a production system by default.

- Instances (nodes) of the WAY4 production system, for example, in setup of "High Availability Cluster" and "High Availability Switch" configurations (see the documents "HA Switch", "WAY4™ HA Cluster"). These solutions are supplied according to separate agreements with the WAY4 vendor.

  Production system nodes are registered in a specialised menu supplied according to a separate agreement. These settings are described in documentation for the corresponding solutions.

This section describes registration of test systems.

To register a WAY4 test system, do as follows:

- Select the user menu item "Full →Configuration Setup →Main Tables → System Instances - Simple". The "System Instances – Simple" form will open (see Fig. 18).

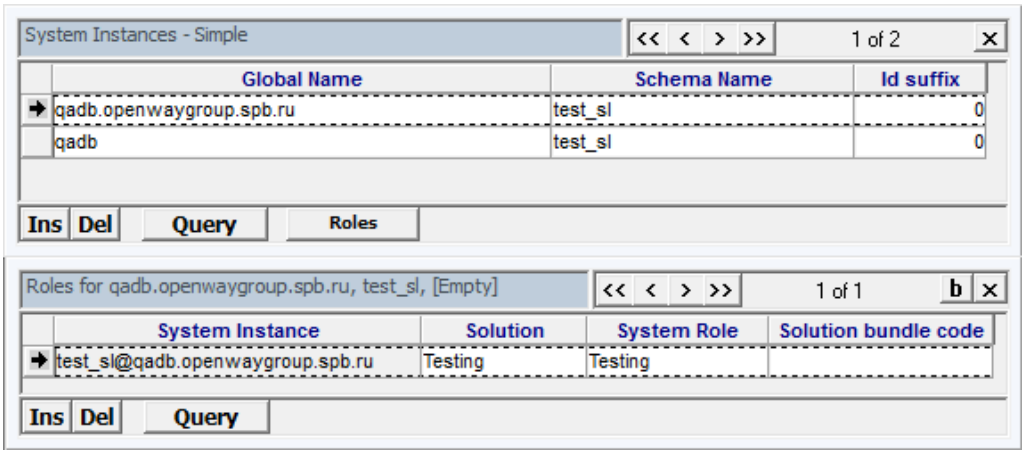


Fig. 18. Registering a WAY4 test system

Fill in the following fields in the "System Instances – Simple" form:

- *Global Name* – schema instance name. Information about the schema instance name can be obtained using the following query:

```
Select lower(global_name) from global_name
```

- *Schema Name* – schema owner name (Oracle Owner); the value must be specified in lower-case letters.
- Do not fill in the *Id suffix* field.

- In the "System Instances – Simple" form (see Fig. 18), click the [Roles] button. In the "Roles for <...>" form that opens, add a new record and fill in the fields. The "Testing" value should be specified in the *Solution* and *SystemRole* fields (see Fig. 18).

The *Solution* field and the *System Role* field that depends on it contain other values, used in particular when registering production system nodes. After registration in a specialised menu, these nodes are shown in the "System Instances – System" form. Possible values for the *Solution* field:

- "Configuration" – configuration replication system (this value is reserved for future use).
- "Data Split" – production sytem with several equal processing nodes (this value is reserved for future use).
- "Data Warehouse" – WAY4 Datamart sytem (this value is reserved for future use).
- "HA Cluster" – WAY4 production system topology. WAY4's high availability is ensured by the use of two and more system instances (nodes) able to process critical operations. For more information, see

the document "WAY4™ HA Cluster". The solution is supplied according to a separate agreement with the WAY4 vendor.

- "HA Switch" – WAY4 production system topology. WAY4's high availability is ensured by the use of two and more system instances (nodes) able to process online operations. For more information, see the document "HA Switch". The solution is supplied according to a separate agreement with the WAY4 vendor.
- "Stand-In" – production system topology with two nodes (for more information, see the document "WAY4 Stand-in").
- "Remote Portal" – value reserved for future use.
- "Remote Testing" – value reserved for future use.

10 "NW Groups" Dictionary

The "NW Groups" dictionary is used to determine rules for a financial institution in WAY4 (member) to interact with various bank and payment networks (Core Banking System, VISA, MasterCard). The list of payment networks (NW_AREA) and sets of parameters specific for interacting with each payment network (the NW_PARM table) are predefined and cannot be edited (see Fig. 19).

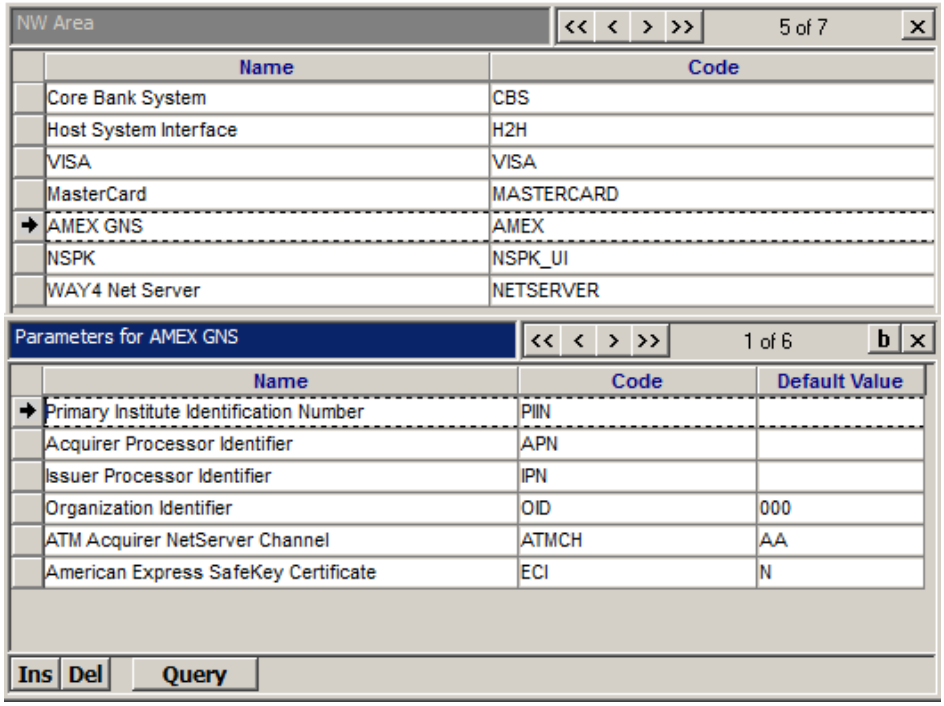


Fig. 19. Set of parameters for interacting with AMEX

One member may interact with different payment systems and have different sets of parameters. Sets of parameters for the interaction of a certain member/members with a payment system are created in the "NW Groups" dictionary, menu item "Full →Configuration Setup →Main Tables →NW Groups". Parameters can be grouped by any formal characteristic. In Fig. 20 parameter for Core Banking System are grouped according to the type of file imported from the CBS.

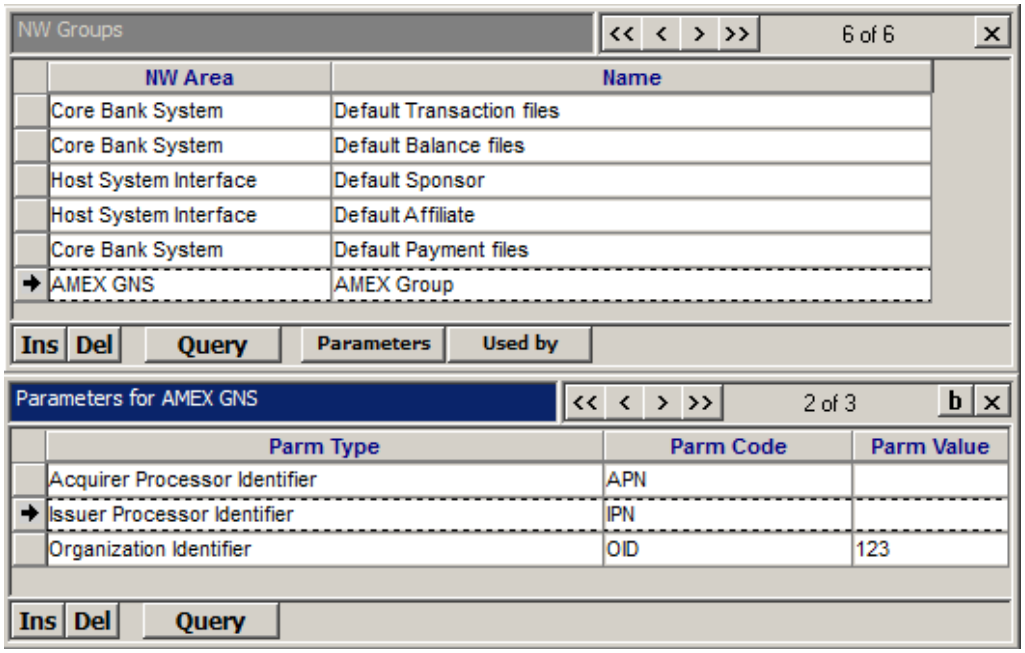


Fig. 20. Set of parameters for the "AMEX Group"

The "NW Groups" form contains the following fields:

- *NW Area* – payment systems with which the member interacts. The value is selected from a predefined list (see Fig. 19).
- *Name* – name of group of parameters for interacting with the payment system. The value is entered manually.

The "Parameters for <parameter group name>" form contains the following fields:

- *Parm Type* – parameter name. The value is selected from a predefined list of parameters for interacting with the payment system.
- *Parm Code* – parameter code. The value is generated automatically.
- *Parm Value* – parameter value. The value is set manually.

To generate a list of member parameters, do as follows:

1. Create a group for interacting with the payment system in the "NW Groups" form.
2. Use the [Parameters] button to open the "Parameters for <group name>" form.
3. Generate a list of group parameters. Parameters from the list predefined for interaction with this payment system can be included in the group.
4. Select the name of the parameter in the *Parm Type* field. The code of the parameter in the *Parm Code* field and the default value of the parameter

in the *Parm Value* field are generated automatically. Set the value of the parameter in the *Parm Value* field.

5. In the "Bank Acquiring Parameters" form define the payment system for which rules for the member's interaction are being set up, menu item "Full →Configuration Setup →Main Tables →NW Groups". Use the [Ins] button to create a new record. Specify the financial institution's ID in the *Acq ID* field. In the *Member ID Type* field specify this member's area of activity (see Fig. 21). This form's fields are described in detail in the section "Configuring Online Interaction Between Affiliated Banks and Payment Systems" of the document "Financial Institutions".

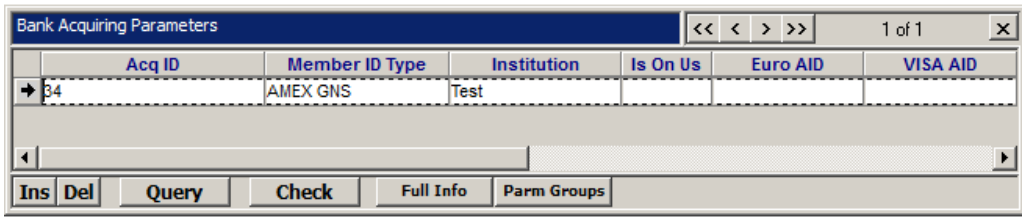


Fig. 21. Setting up member interation with a payment system

6. Assign the member a group of parameters for the selected payment system. To do so, click on the [Parm Groups] button in the "Bank Acquiring Parameters" form to open the "Parm Groups for Bank Accquiring Parameters". In the *NW Area* field, specify the same payment system as in the *Member ID Type* field of the "Bank Acquiring Parameters" form (see Fig. 21). Select the required group of parameters (see Fig. 22).

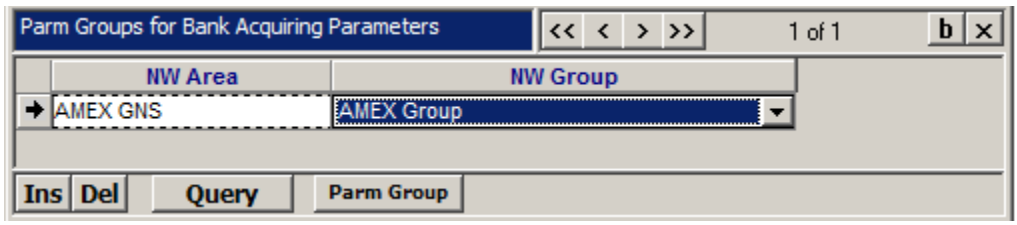


Fig. 22. "AMEX Group" parameter group specifying the member's interaction with AMEX GNS

7. To view participants using the parameter group, click on the [Used by] button in the "NW Groups" form. Members using this parameter group are shown in the "Used by for <group name>" form (see Fig. 23).

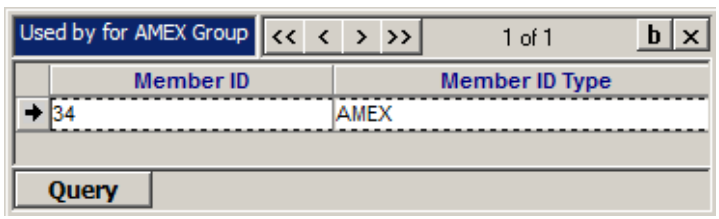


Fig. 23. Member using the "AMEX Group" parameter group for interaction with AMEX GNS

The "Used by for <group name>" form contains the following fields"

- *Member ID* – ID of the financial institution.
- *Member ID Type* – payment system code (*Code* field value in the "NW Area" form, see Fig. 19).

11 "Global Constants" Dictionary

The "Global Constants" dictionary is used to register global constants used in WAY4, menu item "Full →Configuration Setup →Main Tables →Global Constants" (see Fig. 24).

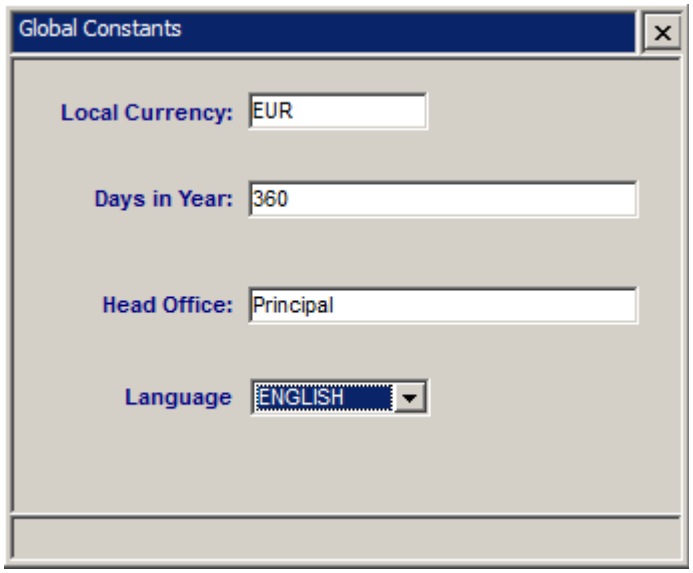


Fig. 24. Values of global constants

The "Global Constants" form specifies the following global constants:

- *Local Currency* – local currency, used when setting up a financial institution.
- *Days in Year* – number of days in a year. Used when calculating interest (see the section "Number of Days in a Year" of the document "Interest Accrual").
- *Head Office* – head financial institution. Used when defining options for entering currency rates (see the section "Set FX Rates" of the document "Daily Procedures") and setting up interbranch routing (see the section "Interbranch Transactions" of the document "Financial Institutions"), etc.
- *Language* – language. Use when setting up software messages, generating reports, configuring client messages, etc. The value is selected from the list generated in the "Languages" table (see ""Languages" dictionary"Languages" dictionary").

The values of global constants can be redefined for a user or user group in the "Constants for <user name/user group name>" form. To open this form, click on the [Constants] button in the "User Groups and Users – View" form ("Full →DB Administrator Utilities →Users & Grants →User Groups and Users - View") (see

the section "Initialising Local Constants" of the document "DB Manager User Management").

12 Redefining System Response Codes

The "Response Codes Customise" dictionary makes it possible to redefine system response codes for authorisation requests if this is a business requirement, for example to redefine response codes according to payment system requirements.

The dictionary is opened with the menu item "Full →Main Tables →Response Code (Customise)" (see Fig. 25).

Response Codes (Customise) << < > >> 2 of 183

Response Code	Level	Message Text
0	0	Successfully completed
1	4	Refer to card issuer
2	5	Refer to card issuer's special condition
3	11	Invalid merchant / source
4	10	PICK UP
5	1	Do not Honour
6	5	Error
7	5	Pick-up card, special condition
8	0	Honour with identification
9	5	Request in progress
10	0	Approved for partial amount
11	5	Approved (VIP)
12	12	Invalid transaction
13	5	Invalid amount

Ins Del Query Add RC Repl.RC

Fig. 25. List of predefined system response codes

The "Response Codes (Customise)" form contains the following fields:

- *Response Codes* – system response code when processing a transaction.
- *Level* – response code importance level. Messages with level "0" correspond to successfully processed requests. Messages with level ">0" are either warnings or error messages. The higher the level, the more critical the error.
- *Message Text* – text description of the response code.

To redefine a system response code:

- Select the code to be replaced.
- Using the [Repl.RC] button, open the "Repl.RC for Response Codes (Customise)" form.

- Using the [Add RC] button in the "Response Codes (Customise)" form, add a new row in the "Repl.RC for Response Codes (Customise)" form (see Fig. 26).
- In the *Channel* field, select the channel for processing transaction information. If no value is defined in this field, the rule will be applied for all channels registered in WAY4 (see the section ""Message Channels" Dictionary").
- Select a transaction condition in the *TransCond* field.
- Select the code for redefinition in the *Response Code* field.

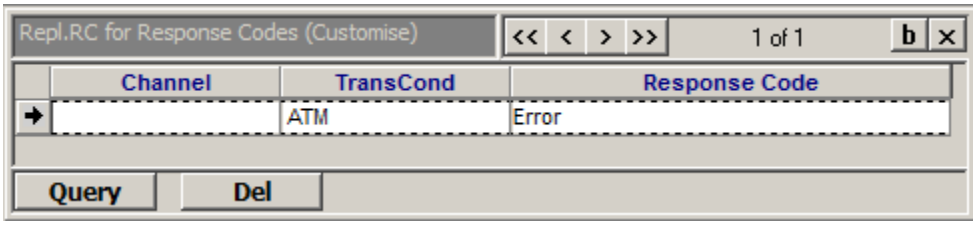


Fig. 26. Redefining the response code for Response Code="1"

13 "Currency Table" Dictionary

The "Currency Table" dictionary is used to register currencies, menu item "Full → Configuration Setup →Main Tables →Currency Table". For more information, see the section "Currency Table" of the document "Currency Conversion".

14 "Bank Acquiring Parameters" Dictionary

The "Bank Acquiring Parameters" dictionary is used to configure parameters for payment system member banks, menu item "Full →Configuration Setup →Main Tables →Bank Acquiring Parameters"ю

For a more detailed description, see the section "Configuring Online Interaction between Affiliated Banks and Payment Systems" of the document "Financial Institutions".

15 Setting workstation numbers for interaction with NetServer/ Transaction Switch

When performing operations like voice authorization, ATM management, etc., a workstation number is used to ensure interaction with NetServer/ Transaction Switch. Workstation numbers are set up, viewed and edited in a form opened by selecting the menu item "Full →Configuration Setup →Merchant Device Setup → OpenWay Stations" (see Fig. 27).

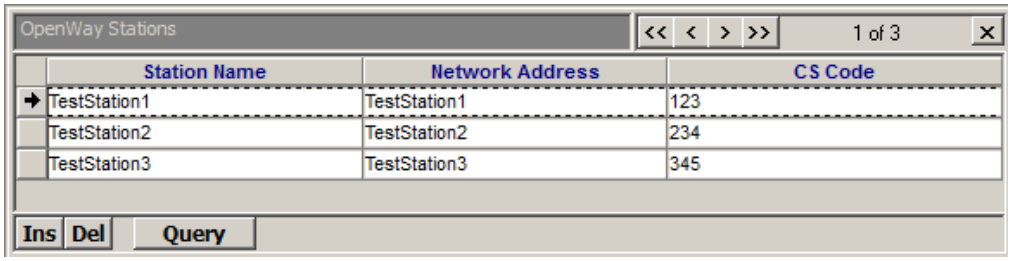


Fig. 27. Form for setting up workstation numbers

It is necessary to specify the following values in the fields of this form:

- *Station Name* – name of the user's workstation (it is permissible to specify the value corresponding to the value of the *NetWork Address* field)
- *NetWork Address* –the workstation's network name, used by the system
- *CS Code* – a unique three-digit workstation code.

For the changes to come into effect, restart the application (DBManager or W4Manager).

16 "Languages" dictionary

The "Languages" dictionary contains a list of languages that can be used in WAY4, for example, when setting up the "Message Dictionary" (see the section "Message Dictionary"), global constants (see the section ""Global Constants" Dictionary"), and local constants for users/user groups (see the section "Initialising Local Constants" of the document "DB Manager User Management").

The list is opened using the menu item "Full →Configuration Setup →Languages and Localisation →Languages", see Fig. 28.

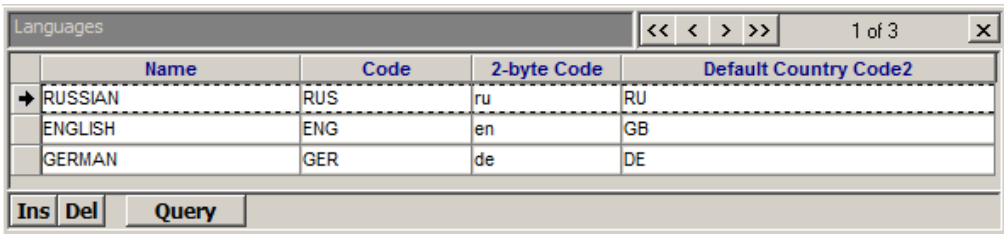


Fig. 28. "Languages" dictionary

The form contains the following fields:

- *Name* – language name.
- *Code* – code of the language used in WAY4.
- *2-byte Code* – two-letter language code; the field's value must comply with the ISO 639-1 international standard for language codes.
- *Default Country Code2* – two-letter country code; the field's value must comply with the ISO 3166-1 alpha-2 international standard for country codes.

For an example of language setup for reports, see the section "Local Language Support in Report Generation" of the document "Generating Reports in WAY4™".

17 "Address Types" dictionary

The "Address Types" dictionary contains a list of address types. Address types allow an address to be classified depending on its purpose. For example, an address may be used for delivering statements, or when printing PIN mailers. In addition, conditions set for each address type determine rules for generating client and contract address information and rules to search for a relevant client/contract address.

The dictionary is opened with the menu item "Full →Configuration Setup →Client Classifiers →Address Types", see Fig. 29.

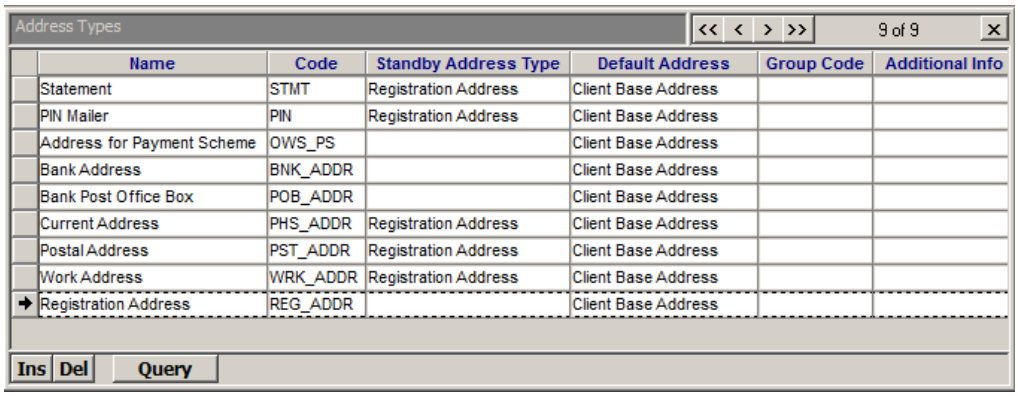


Fig. 29. "Address Types" dictionary

The form contains the following fields:

- *Name* – address type name.
- *Code* – address type code used in WAY4.
- *Standby Address Type* – address type for which a search for an active address is made, if an active address of the type specified in the *Name* field is found neither for the client record, nor in the client contract tree.
- *Default Address* – rule to search for an address of the specified type:
  - "Client Base Address" – the address registered in the client record (in the Client table) is used.
  - "Contract Address Only" – a search for an active address is only made for the contract.
  - "Branch Address" – if an active address of this type is found for neither the client record or in the client contract tree, an address of this type, set for the bank branch, will be used.
  - None – a hierarchical search is made for an active address of this type along the contract tree to the higher-ranking contract.

The value of the *Default Address* field can be redefined using the DEFAULT_ADDRESS tag for a financial institution (in the *Special Parms* field of the form for the financial institution, menu item" Full →Configuration Setup → Main Tables → Financial Institutions → [Details]"). The DEFAULT_ADDRESS tag may have the following values:

- "N" – do not use a default address.
- "Y" – use the default address from the client's record (from the Client table).
- "B" – use the default address for the bank branch.
- "C" – a search for an active address is only made for the contract.

- *Group Code* – rule for generating address informaiton for field groups (either in the CLIENT table, or in the CLIENT_ADDRESS table).
  - By default (the *Group Code* field is not filled in) – one of the fields from the *Address Line* group must be filled in, or *Email* or *Add Info.*
  - "Contract Data" – one of the fields from the *Phone* group must be filled in, or *Email.*
  - "Custom Check" – address information is not mandatory for client records or client contracts.
  - "Email" – *Email* must be specified.
  - "Postal Address" – one of the fields form the *Address Line* group must be filled in.
- *Additional Info* – additional information.

17.1 Address Search

If the "Base Address" field group for a client record in the "Client" table is not filled in, addresses set for the client or contract are stored in the "Client addresses" table and linked with either a client record or client contract.

A search for an address of the corresponding type is made in the CLIENT table and the CLIENT_ADDRESS table.

The search is made for an active address of the required type (ADDRESS_TYPE field in the CLIENT_ADDRESS table) for the contract (for example, for which an Event is opened). If an address for the contract is not found, the value of the *Default Addresses* field of the "Address Types" dictionary is checked for this address type (USE_CLIENT_DEFAULT field in the ADDRESS_TYPE table), see Fig. 29. If the *Default Addresses* field has the value:

- "Client Base Address" – the address is searched for in the client record (CLIENT table).
- "Contract Address Only" – a search is made for an active address only under the contract.
- None – a hierarchical search is made along the contract tree down to the "root".

In each level, the presence of an address of the requested type for the contract is checked, and depending on the value of the *Default Addresses* field, for this contract's client. If an address is still not found at the "root" level, depending on the value of the *Default Addresses* field, the search can be continued in the "Liability" tree according to the same algorithm.

If an active address is found in the CLIENT_ADDRESS table, but a different address type is set in the *Use Adr* field of the found record (COPY_TO_ADDRESS field of the "Client address" table), a search is made for a new address of the specified type according to the same algorithm.

If an address is not found, a search for an active address is made for the address type specified in the *Standby Address Type* field of the "Address types" dictionary according to the same algorithm.

In the current implementation, the list of a client's addresses does not include the address that is specified in the client's record (address information from the CLIENT table). If for some reason, the address from the client record must be used, set the value of the USE_CL_BASE_ADDR global parameter to "Y".

17.2 Address transliteration rules

WAY4 supports transliteration of addresses into English.

Information specified in the FIRST_NAM, LAST_NAM, CITY, MUNICIPALITY_CODE, STATE, ADDRESS_LINE_1, ADDRESS_LINE_2, ADDRESS_LINE_3, and ADDRESS_LINE_4 fields of a client or contract record is transliterated.

The following affect transliteration rules:

- Client or contract language; set in the Language field when registering the client or contract (Language field of the CLIENT_ADDRESSES table).
- Language specified in the system's global settings ("Full →Configuration Setup →Main Tables →Global Constants").
- System user's language; set in the Language field of the "Constants for <user group name>" form that is opened by clicking the [Constants] button in the "User Groups and Users - View" grid form ("Full →DB

Administrator Utilities →Users & Grants →User Groups and Users – View").

- Values of the USE_CL_BASE_ADDR and USE_SYSTEM_ADDR_LANG global parameters.

Transliteration rules:

The language of an address is determined from the table with address information for a client and the client's contracts (value of the *Language* field in the Client_Address table).

If no language is specified for a client or contract, the language of the address will be the one that is specified in the system's global parameters.

If the language of an address differs from the user's language and is not English, it is transliterated. If the language of the address and user are the same, there is no transliteration. If the language of the address or the user's language cannot be determined, there is no transliteration.

If for some reason it's necessary to use the address (and address language) from the client's record (from the Client table), the following settings are required:

1. The USE_CL_BASE_ADDR global parameter value must be "Y".
2. The USE_SYSTEM_ADDR_LANG global parameter value must be "N".
3. The "Address_Type" dictionary must contain a rule to search for an address of the specified type using the address that is registered in the client's record (for addresses of the corresponding type, the value of the Default Address field in the "Address_Type" dictionary must be "Client Base Address").

If the address in a client's record is not English, the address is transliterated. If an address is not specified in the client's record, the system language is considered to be the language of the address.

18 Additional settings for Merchant Service Workbench

Merchant Service Workbench (MSW) is a solution that provides access to WAY4 Cards through a web client.

MSW is delivered to OpenWay through acquired software licenses.

To show total payments to a merchant in MSW if reimbursement for retail transactions is credited to a contract in WAY4, configure the "Merchant Payments Mapping" dictionary, menu item "Full → Products → Acquiring Products →Merchant Payments Mapping".

In configurations where payments to merchants are made to accounts that are not kept in WAY4, setup of this dictionary is optional, but recommended.

This dictionary is configured so that technical documents (for example, technical entries for moving funds in a contract hierarchy) are excluded and the user interface shows only total payments for financial documents that are generated when standing payment orders are processed. The list of transaction types for these standing payment orders should be listed in the "Merchant Payments Mapping" dictionary (see Fig. 30).

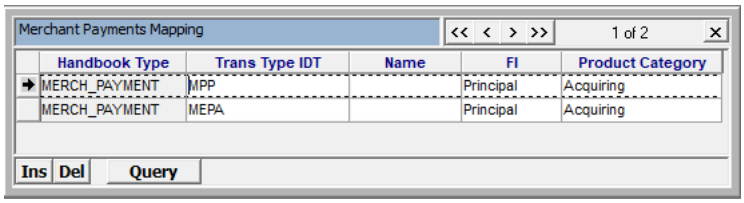


Fig. 30. Example of transaction type dictionary

"Merchant Payments Mapping" form fields:

- *Handbook Type* – dictionary code. The value is set in the global parameter API_MERCH_PAYMENT_GROUP. The default value is "MERCH_PAYMENT".
- *Trans Type IDT* – the value of the *Trans Type IDT* field in the "Transaction – ALL" form (TRANS_TYPE table), menu item "Full →Configuration Setup → Transaction Types →Transaction – ALL".
- *Name* – transaction type name.
- *FI* – financial institution.
- *Product Category* – Product category. The default value is "Acquiring".