The mapping module

July 2011

Contents

1. Introduction

The aSISt base module is used for data entry and XBRL reports reconciliation and submission. The general task of aSISt is to generate XBRL reports, which is a final stage in the reporting process.
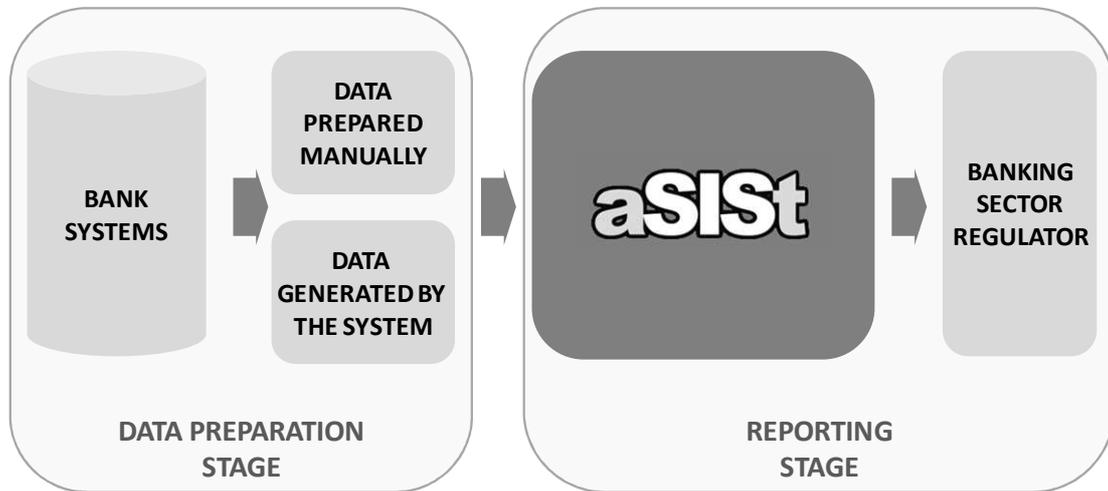
Picture 1. Standard scheme of the reporting process

The base module has a number of interfaces which enable the user to import data from previously prepared sets. However, the structure of the sets must be strictly compatible with the structure of the report:

- the structure of the sets has to match exactly the structure of the report,
- each cell in the report must have only one assigned position in the previously prepared data source.

The attempt to meet these requirements is both an arduous and time-consuming task. Moreover, the process of collecting data from bank systems consists very often of several stages.
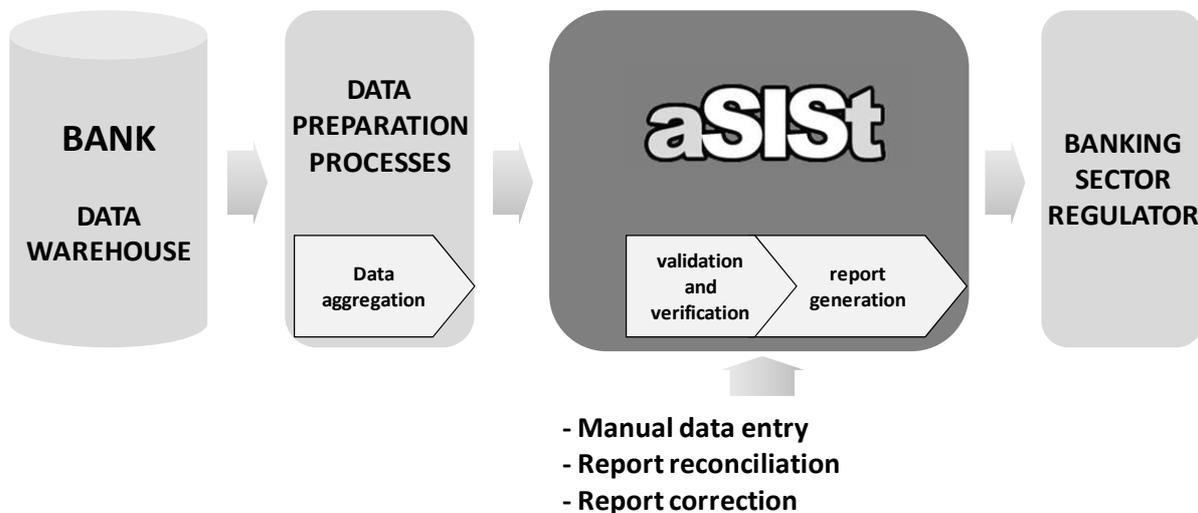
Picture 2. Reporting process – current situation

Long processing time is not the only disadvantage of manual data preparation. Processing has to be repeated for each reporting cycle.

 A suitable solution to this problem is **the MAPPING MODULE** - an additional module in aSISt.

The mapping module provides a mechanism for automatic calculation of values from multiple sources.
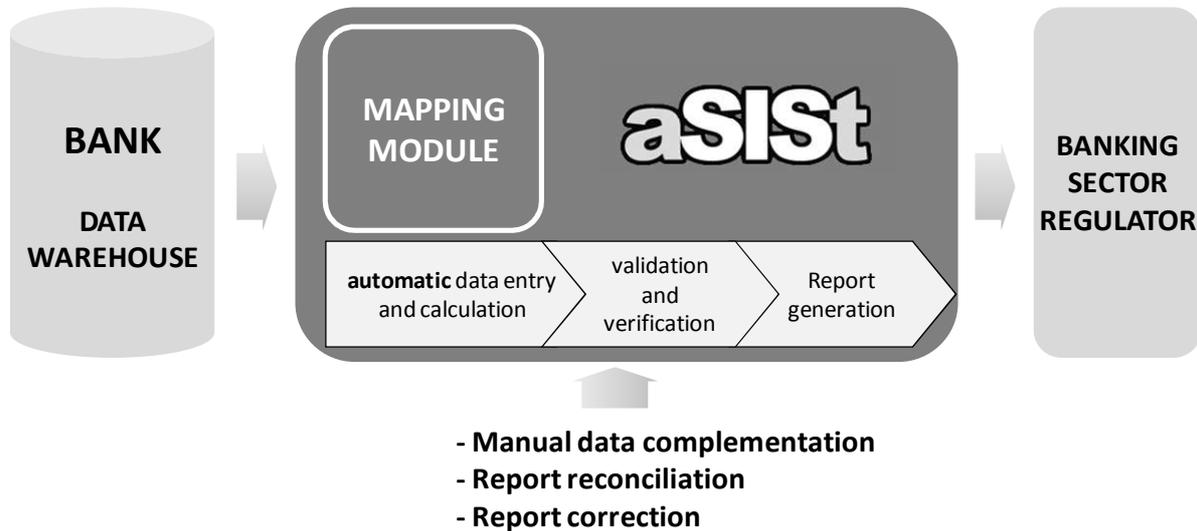
Picture 3. Reporting process –aSISt with the mapping module

Owing to the mapping module, aSISt provides a variety of flexible ways of data processing which require no adjustments to the form and shape of the data sources. The user saves a lot of time and avoids additional preprocessing and aggregation of data.

2. Automation of data entry with the mapping module

An alternative to importing data from a file is a method of automatic and active data fetching, and its calculation and entry with use of the mapping module. The advantage of this solution is the automation of several operations which ordinarily would have to be carried out in Excel or manually.

The mapping module consists of several components:
- Data sources management
- Mappings management
- Mappings editor
- Calculation process

All of them are described in details below.

Data sources management

Data sources management is used for defining the sources of data available for mappings.

As a data source one may regard any table with data containing adequate information that may be used to calculate values in the report.

Mappings management

This module allows to define many sets of mappings. The user may employ several sets simultaneously to:
- manage different types of mappings,
- aggregate data from multiple sources – e.g., different systems during accounting systems replacement,
- consolidate reports – map the reports and the consolidation adjustments.

Mappings editor

The mappings editor is one of the integral components of the mapping module. It allows to save, edit and delete the mapping formulas as well as to export and import the mapping sets to and from external files.

Calculation process

Fetching data by mappings is an additional function in aSISt. It initiates the process of automatic calculation and entering values into the report cells. The mapping module provides similar variants of values entry as the importing function does – it means that the user may, e.g., overwrite all mapped cells, overwrite only the empty cells, etc. The difference is that in the mapping module **one cell may be calculated from many records of one source as well as from many sources simultaneously.**

The benefits of mapping module

The mapping module, owing to its optimally adjusted functionality, offers a variety of benefits:

- may be used both as a basic or an additional data entering tool – the mapping module is very functional and may be applied to all reports (taxonomies), both to entire reports as well as to their specific parts,
- provides the quickest method of data entry from sources generated straight by banking systems,
- enables aggregation of data from many sources to one cell in the report, and immediate calculation of its value
- precisely selects from data sources only the values that exactly match the conditions specified by the user and therefore eliminates the possibility of inconsistencies and data loss which may occur by mistake in manual processing - the mapping language allows to use many aggregate functions and selectors at the same time,
- automatically calculates every report cell selected by the user and thus helps to avoid many mistakes which occur in manual calculations,
- provides standardization of the process of automatic data entry into the report – the
- mapping module is a specialized tool designed particularly for this purpose,
- gives a possibility of introducing changes to once defined mappings by overwriting,
- transforming, adding and/or erasing the single mappings from the set – the editing process is simple and may be carried out in text formats in the module window as well as in text editors,
- suitable also for data entry and calculation in list forms,
- saves time and costs of the reporting process.

3. Terms and examples

Report cell types – primary and calculated cell
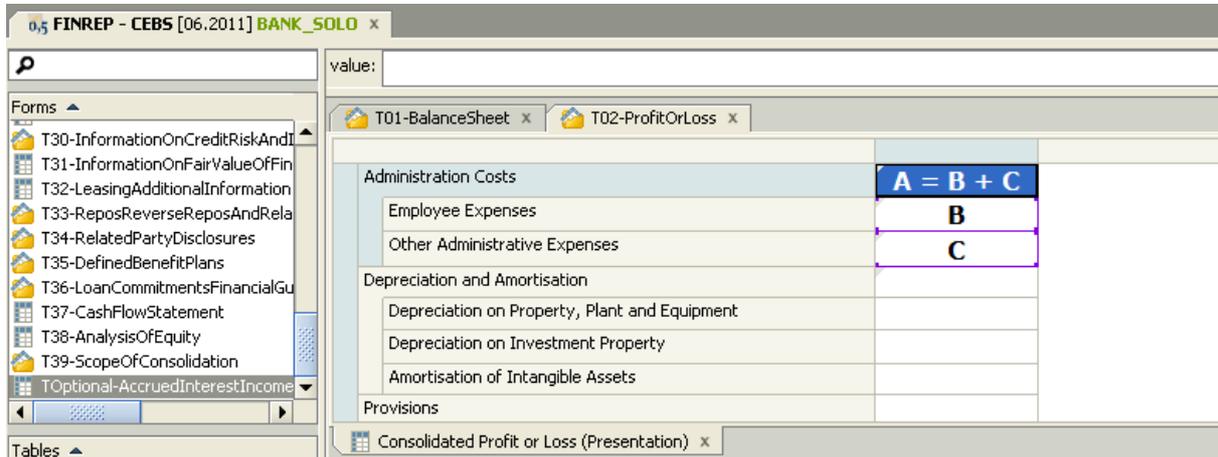
Example 1. Primary cells (independent) and calculated cells (dependent)

Primary cells: B and C

Calculated cell: A

Primary cells have to be entered by the user. Alternatively, they may be calculated automatically from external data sources with the mapping module (with a formula defining the source and specifying values for the cell).

Calculated cells are defined by validation rules and calculations (e.g., components sum), they do not have to be entered by the user as they are calculated automatically.

Mapping data from sources to the report cells

Every cell in the report has a unique ID determined by XBRL format which is required by the banking sector regulator.
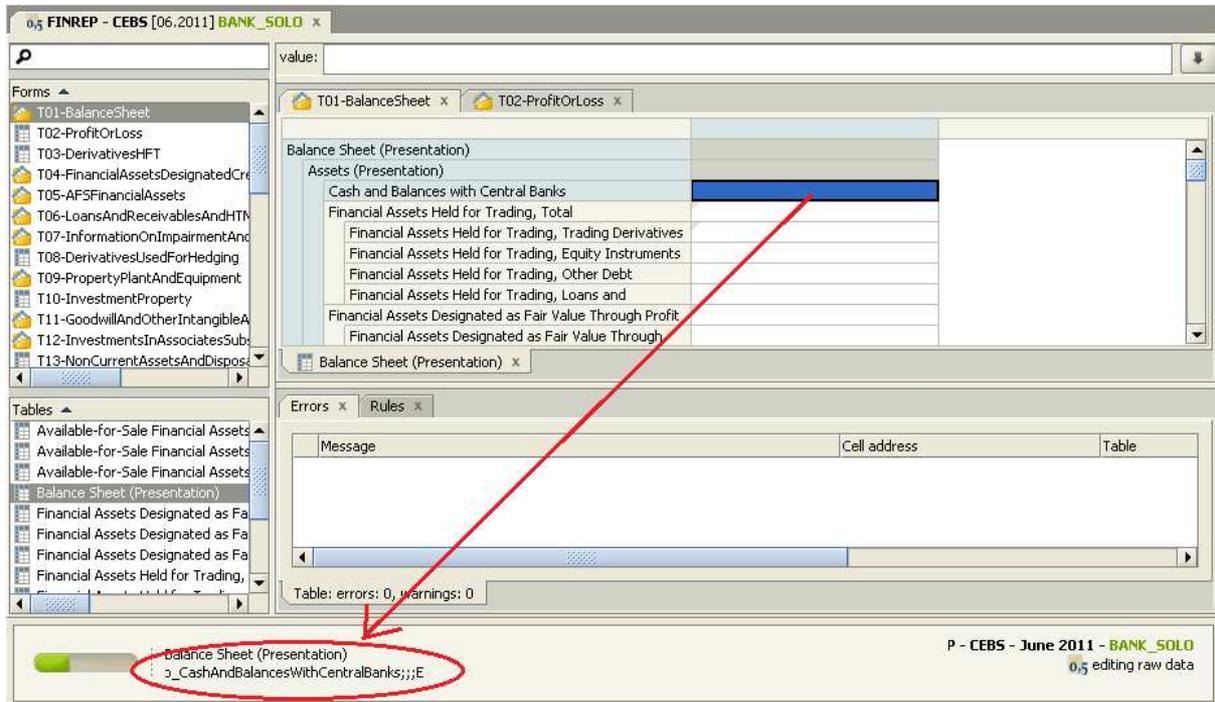

Example 2. Cell ID in the report

However, banks in their systems use different types of attributes, such as chart of accounts, numbers, accounts attributes, customers attributes, etc.

Mapping is, more or less, a link between a specified set of attributes of a data source and a cell in the report.

The significance of the ID in the reporting process is not important for aSISt users. The task of linking attributes with cells and cells with the ID is taken over by aSISt.

The mapping module presents mappings in the report table form, just as in the aSISt base module. This makes their creation much easier and their management and edition more accessible.
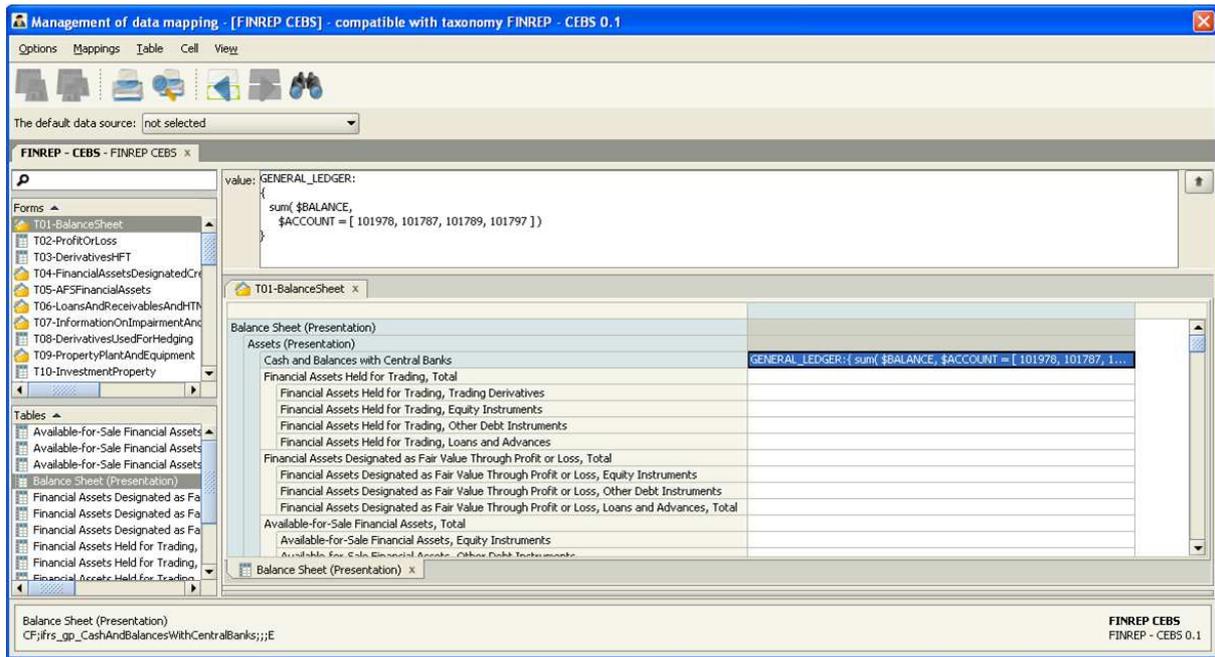
Example 3. The mapping module window with exemplary mappings

Data sources for mappings

Mapping fetches data from any source the structure of which matches the criteria listed below:

1. The source is a table, which means that all records (rows) have the same structure and the data source is a combination of the records.
2. The table has a headline row which identifies the columns of the table.
3. The data source is in one of the following formats: CSV, file XLS [MS Excel format] or a
4. Table which is a result of SQL query

Example 4. Exemplary source of data GENERAL_LEDGER

The table which is the source of data may be composed of any content, e.g.:

- the table may contain unlimited number of rows and columns
- columns and rows may have any name
- the source file may have any name

The source may then be any table created in any way, e.g., a table exported from the main system or created manually.

Mapping language

The essential element of mapping language is a mapping expression as in the form presented below:

SOURCE_NAME : {
  aggregates function (calculation on the selected records of data , selector) }

where:
- source name – defines a table which contains required data
- aggregates function – an arithmetical operation on a series of records, e.g. "sum"
- record of data – a row in the table of data
- selector – filter which applies the conditions to data selection

The columns of the data source are indicated by a special mark "$".

A simple mapping example:

GENERAL_LEDGER : { sum ($REVISION , $ACCOUNT = [101978, 101787]) }

Which reads as follows:
- for the data source GENERAL_LEDGER
- it aggregates (sum)
- the values from the REVISION column
- for records which in the ACCOUNT column have numbers 101978 and 101787

Mapping expressions link the attributes of the report with expected calculation of the data set selected from the source. Summarizing, mapping expressions describe mappings.
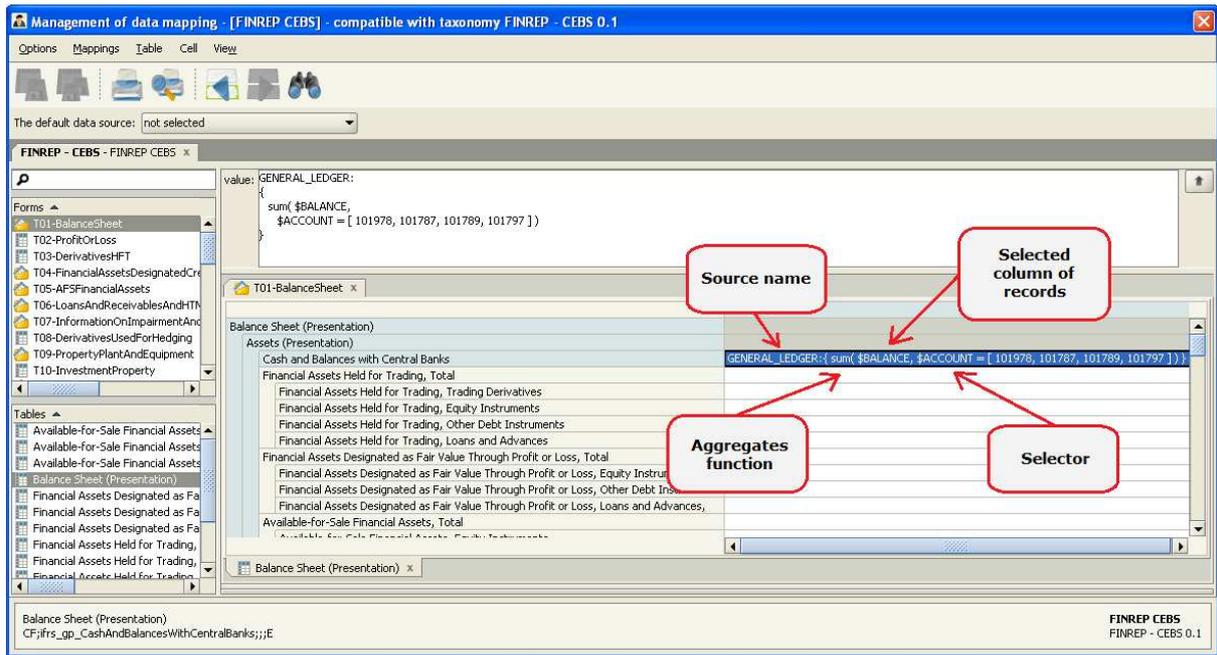
Example 5. Structure of mapping expression

Another example of a calculation of data records:

> **GENERAL_LEDGER : { sum ($BALANCE - $REVISION , $ACCOUNT = [101978, 101787]) }**

The above means that for records which in the ACCOUNT column have numbers 101978 and 101787 are aggregated values in BALANCE column minus values in REVISION column.

Mapping allows one to use many conditions in one expression and to combine them with one of the logical operators, for example:

> **AND**   -   condition1   **&&**   condition2
>
> **OR**   -   condition1   **||**   condition2

This may be useful in a situation when a few conditions need to be applied to one source of data.

For the exemplary data source presented below, there are listed values (the VALUE column) of revision of several accounts (the ACCOUNT column) within several months, and it is indicated whether they have been accepted or not for accounting (the ACCEPTANCE column).

	A	B	C	D	E	F
1	CODE	ACCOUNT	NAME	VALUE	ACCEPTANCE	
2	Rev/CHF/February/01	101797	change	1 000	YES	
3	Rev/GBP/February/02	101789	change	4 000	YES	
4	Rev/EUR/April/01	101978	change	2 000	YES	
5	Rev/EUR/May/01	101978	change	12 000	YES	
6	Rev/EUR/May/02	101978	change	8 000	YES	
7	Rev/GBP/May/03	101789	change	3 000	YES	
8	Rev/EUR/May/04	101978	change	6 000	NO	
9	Rev/EUR/September/01	101978	change	-1 000	NO	
10	Rev/USD/September/02	101787	change	2 000	NO	
11						

**Example 6. Exemplary source of data – REVISION**

Exemplary situation - to obtain the value of revision for EUR (account 101978) which has been already accepted, the mapping expression needs to include two conditions:

REVISION : { sum($VALUE , $ACCOUNT = 101978 && $ACCEPTANCE = YES) }

One mapping may include any number of data sources combined. That allows the user to calculate one cell in the report from many records of one source as well as from many records of many sources.

For example (tables 4 and 6) - to obtain the value for EUR 'account 101978' reduced by 'accepted for accounting revisions', the mapping would be as follows:

GENERAL_LEDGER : { sum ($BALANCE , $ACCOUNT = 101978) }
- REVISION : { sum($VALUE , $ACCOUNT = 101978 && $ACCEPTANCE = YES) }
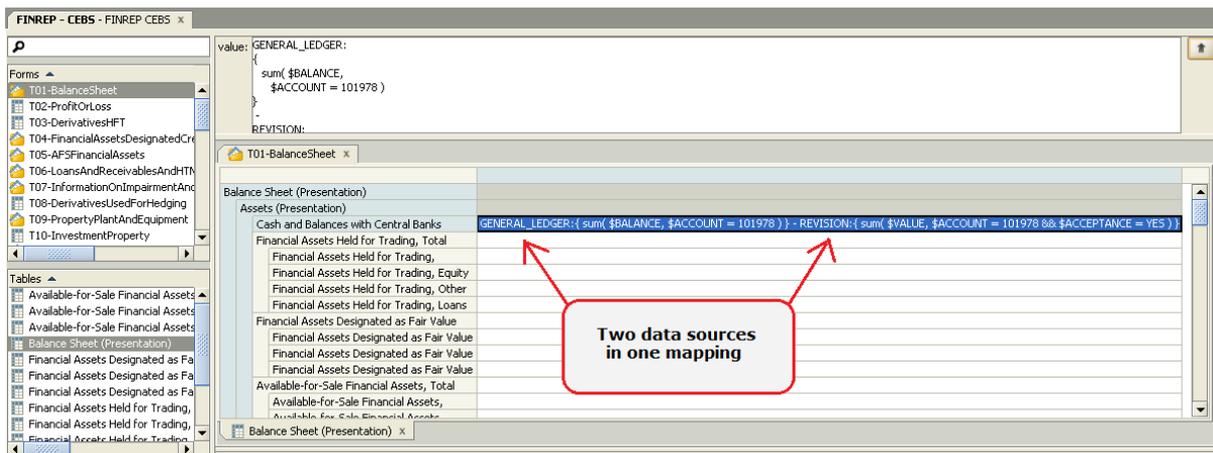
Example 7. Mapping from two separate data sources

And the result for the tables from examples 4 and 6 are the following:

150 000 – (2 000 + 12 000 + 8 000) = 128 000
(balance) –   (revision)

Example 8. The result of mapping data from two different data sources (tables 4 and 6)

For more information on business or technical matters, please contact directly:

Piotr Malczak
Mobile: +48 515 128 933
E-mail: Piotr.Malczak@gpm-systemy.com