

AX003R.MBR

Path: NXCLOUD/rpgsrc/AX003R.MBR **Generated:** 2026-01-08 12:58:02 **Processing Time:** 15083ms

Business Logic for XML Generation from Financial Records

This document outlines the business rules that govern the XML generation process for financial records, based on an analysis of the RPG program AX003R. The primary focus is on how the program retrieves financial data and formats it into XML for output.

The core logic for XML generation is contained within the SXMLUT subroutine in AX003R. The program processes financial records from the RHTRPF file and generates corresponding XML output based on specified criteria.

Order Status and Header Rules

XML Generation: RHTRPF, AXMLUT

1. Firm Selection

- **Logic:** The program sets the firm number to be processed based on the input parameters.
- **File:** RHTRPF (Financial records file)
- **Field:** rhrl5_firm
- **Condition:** The firm number is set to the variable w_firmn at the beginning of processing.

2. Record Filtering

- **Logic:** The program filters records based on the firm, year, and period.
- **File:** RHTRPF
- **Fields:** rhrl5_raar, rhrl5_rper
- **Condition:** Records are processed only if the year is less than or equal to the specified end year and the period is less than or equal to the specified end period.

Configuration and Authorization Rules

1. XML Output Initialization

- **Logic:** The program initializes XML output variables before processing records.
- **Files:**
- AXMLUT (XML output file)
- **Fields:**
- XML_Output
- **Condition:** The XML output is initialized at the start of the program and updated throughout the processing of records.

2. Record Read Logic

- **Logic:** The program reads records from the RHTRPF file in a loop until all relevant records are processed.
- **File:** RHTRPF
- **Field:** rhrl5_biln
- **Condition:** The loop continues until the end of the file is reached or until a specific condition is met.

Financial and Transactional Rules

1. Transaction Processing

•**Logic:** Each transaction is processed to generate XML lines based on the record data.

•**File:** RHTRPF

•**Fields:**

•rbbiln (Bill number)

•rbbdat (Bill date)

•**Condition:** Transactions are processed only if they match the current bill number and date being processed.

2. Final XML Output

•**Logic:** The program generates the final XML output after processing all records.

•**File:** AXMLUT

•**Condition:** The final output is written to the XML file once all records have been processed.

Special Conditions (Program-Specific)

1. Error Handling in XML Generation (AX003R)

•**Logic:** The program includes error handling to skip lines that do not meet certain criteria for XML output.

•**File:** AXMLUT

•**Field:** axmaop

•**Condition:** Lines marked with 'X' are skipped and not included in the XML output.

2. Currency Handling (AX003R)

•**Logic:** The program processes currency values and generates corresponding XML entries for currency codes and amounts.

•**File:** RHTRPF

•**Fields:** rbvalk (Currency code), rbvalb (Currency amount)

•**Condition:** Currency information is included in the XML output if the values are valid and not equal to zero.

Subprogram Calls Affecting Logic

Beyond direct file checks, several external subprograms are called that play a significant role in the workflow.

1. SXMLUT (XML Line Output)

•**Trigger:** Called during the processing of each financial record.

•**Logic:** This subroutine formats and writes XML lines based on the current record data.

•**Impact:** This call acts as a **major logical gateway** for generating XML output from the processed records.

2. BehKonto (Account Processing)

•**Trigger:** Called for each account associated with a transaction.

•**Logic:** This subroutine processes all lines for a given account and generates the corresponding XML output.

- Impact:** This represents the detailed processing of account transactions, ensuring all relevant data is captured in the XML.

3. redefnum (Number Formatting)

- Trigger:** Called to format numeric fields before output.

- Logic:** This subroutine ensures that numeric values are correctly formatted for XML output.

- Impact:** This is a **destructive filtering step** that ensures only valid and correctly formatted numbers are included in the XML.