

RV001R.MBR

Path: NXKORR/rpgsrc/RV001R.MBR **Generated:** 2026-01-09 10:13:44 **Processing Time:** 12733ms

Business Logic for RV001R

This document outlines the business rules that govern the process of reading lines from the FOVF file and converting them into a comma-separated format, based on an analysis of the RPG program RV001R. The primary focus is on the transformation and validation of data as it is processed.

The core logic for data transformation is contained within the control_init subroutine in RV001R. This subroutine initializes variables, reads records from various files, and performs necessary checks and transformations before writing the output.

Order Status and Header Rules

RV001R: FOVF, SOHEL1, VOTYL1, AFORL1, RB10L1, RKUNL1, RB00L1

1. Firm Check

• **Logic:** The process will skip further processing if the current firm does not match the expected firm.

• **File:** FOVF (Input file for reading data)

• **Field:** l_firm

• **Condition:** "The process will not select a record if l_firm is not equal to fffirm."

2. Blank Initialization

• **Logic:** Initializes various fields to blank if the firm check passes.

• **File:** FOVF

• **Fields:** d_dati, d_fdati, w_lnavn, w_lnav, wknavu, w_posm, w2fnav, wkalfa, w2navn, w_ladr1, w_ladr2, w_lposn, w_lposs

• **Condition:** "All specified fields are set to blank upon successful firm check."

Configuration and Authorization Rules

1. Tax Code Assignment

• **Logic:** Assigns the appropriate tax code based on the input data and checks if it needs conversion.

• **Files:**

• RB00L1 (Tax code mapping)

• **Fields:**

• ffdfmom (Input tax code)

• w_mkod (Output tax code)

• **Condition:** "If ffdfmom is blank, w_mkod is assigned from ffcmom; otherwise, it retains ffdfmom."

2. Invoice Type Conversion

• **Logic:** Checks if the invoice type should be converted based on the mapping in RB10L1.

• **File:** RB10L1 (Invoice type mapping)

• **Field:** ffblk

- **Condition:** "The invoice type is checked against RB10L1 to determine if it exists and if conversion is necessary."

Financial and Transactional Rules

1. Data Formatting for Output

- **Logic:** Constructs a comma-separated string from various fields for output.

- **File:** RV01PF (Output file)

- **Fields:**

- ffbiln (Invoice number)

- fftext (Invoice text)

- ffbelp (Line amount)

- **Condition:** "The output string is built from multiple fields, ensuring proper formatting and trimming."

2. Mail Address Handling

- **Logic:** Includes the customer's email address in the output if available.

- **File:** RKUNL1 (Customer data)

- **Condition:** "The email address is appended to the output string if it exists."

Special Conditions (Program-Specific)

1. Data Retrieval from Historical Invoice

- **Logic:** Retrieves data from historical invoices and customer records as needed.

- **File:** SOHEL1, RKUNL1

- **Field:** solkun

- **Condition:** "The process checks for existing customer records and retrieves relevant data if solkun is not zero."

2. Handling of Special Characters

- **Logic:** Translates special Norwegian characters to ensure proper output formatting.

- **File:** RV01PF

- **Condition:** "Special characters are replaced before writing to the output file."

Subprogram Calls Affecting Logic

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

1. AK710R (Routine for handling customer data)

- **Trigger:** Called when processing customer-related data.

- **Logic:** Retrieves additional customer information based on provided parameters.

- **Impact:** "This call enhances the data quality by ensuring all relevant customer details are included."

2. control_init (Initialization subroutine)

- **Trigger:** Called at the beginning of the main processing logic.

- **Logic:** Initializes key variables and prepares the program for data processing.

- **Impact:** "This is a critical setup step that ensures all necessary variables are ready for use."

3. sohel1_key (Key for invoice header lookup)

- Trigger:** Used to set the key for reading invoice header records.
 - Logic:** Establishes a key for accessing the invoice header based on firm and invoice number.
 - Impact:** "This key setup is essential for accurately retrieving invoice header data."
- This documentation provides a comprehensive overview of the business logic implemented in the RV001R program, detailing the rules and processes that govern the transformation of input data into the desired output format.