RV001R.MBR

Path: NXKORR/rpgsrc/RV001R.MBR **Generated**: 2026-01-21 15:43:13 **Processing Time**: 19453ms

Business Logic for RV001R

This document outlines the business rules that govern the processing of invoice data from the FOVF file into a comma-separated format, based on an analysis of the RPG program RV001R. The primary focus is on how invoice lines are read and transformed into a structured output.
The core logic for reading and processing invoice data is contained within the control_init subroutine in RV001R. This subroutine initializes various fields and checks conditions to ensure the integrity of the data before writing it to the output file.

Order Status and Header Rules

RV001R: FOVF, SOHEL1, VOTYL1, AFORL1, RB10L1, RkunL1, RB00L1

1. Firm Check

•**Logic:** The program checks if the current firm (l_firm) matches the firm from the file (fffirm). If not, it skips processing.
•**File:** FOVF (Invoice Data)
•**Field:** l_firm
•**Condition:** The process will not select a record if l_firm is not equal to fffirm.

2. Invoice Date Handling

•**Logic:** The program initializes date fields to blank if the firm check fails.
•**File:** FOVF
•**Field:** d_dati, d_fdati
•**Condition:** If l_firm does not match fffirm, date fields are set to blanks.

Configuration and Authorization Rules

1. VAT Code Assignment

•**Logic:** The program assigns a VAT code based on whether the existing VAT code (ffdmom) is blank or not.
•**Files:**
•FOVF (Invoice Data)
•RB00L1 (VAT Code Reference)
•**Fields:**
•w_mkod (Current VAT Code)
•ffdmom (Existing VAT Code)
•**Condition:** If ffdmom is blank, w_mkod is set to ffcmom; otherwise, it is set to ffdmom.

2. Invoice Type Conversion

•**Logic:** The program checks if the invoice type needs to be converted based on the ffbilk field.
•**File:** RB10L1 (Invoice Type Reference)
•**Field:** ffbilk
•**Condition:** If the invoice type exists in RB10L1, it assigns the corresponding code to xxbilk.

Financial and Transactional Rules

1. Customer Data Retrieval

•**Logic:** The program retrieves customer data from the RkunL1 file if the customer ID is not zero.

•**File:** RkunL1 (Customer Data)

•**Fields:**

•rkunl1_kund (Customer ID)

•w_knavn (Customer Name)

•**Condition:** If solkun is not zero, the program retrieves the customer data.

2. Invoice Line Formatting

•**Logic:** The program formats invoice line data into a comma-separated string for output.

•**File:** RV01PFR (Output File)

•**Condition:** The formatted string includes various fields such as invoice date, amounts, and customer details.

Special Conditions (Program-Specific)

1. Handling of Special Characters (RV001R)

•**Logic:** The program translates special Norwegian characters to their respective codes before writing to the output file.

•**File:** RV01PFR (Output File)

•**Field:** nsdata

•**Condition:** Special characters like 'Æ', 'Ø', and 'Å' are translated during the write operation.

2. Invoice Number Initialization (RV001R)

•**Logic:** The program initializes the temporary invoice number field (t1kkid) to blank at the start of processing.

•**File:** RV01PFR

•**Field:** t1kkid

•**Condition:** This initialization occurs at the beginning of the processing loop.

Subprogram Calls Affecting Logic

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

1. AK710R (Customer Data Retrieval)

•**Trigger:** Called when specific conditions regarding customer data are met.

•**Logic:** This subroutine retrieves additional customer information based on the customer ID.

•**Impact:** This call enriches the invoice data with customer details.

2. control_init (Initialization Subroutine)

•**Trigger:** Called at the beginning of the processing loop.

•**Logic:** Initializes various fields and checks for the presence of required data.

•**Impact:** This is a **critical initialization step** that sets up the environment for processing invoice lines.

3. write (Output Subroutine)

- **Trigger:** Called at the end of processing each invoice line.
- **Logic:** Writes the formatted invoice line to the output file.
- **Impact:** This represents the **final step** in the data transformation process, ensuring that all necessary information is captured in the output format.