

RX000R.MBR

Path: NXCLOUD/rpgsrc/RX000R.MBR **Generated:** 2026-01-08 12:59:01 **Processing Time:** 11475ms

Business Logic for RX000R

This document outlines the business rules that govern the retrieval of file and path information for an FTP job, based on an analysis of the RPG program RX000R. The primary focus is on the logic that determines how the program retrieves and validates necessary parameters for file processing. The core logic for retrieving file and path information is contained within the main program logic. The program checks for the existence of records in specific files and sets parameters accordingly.

Order Status and Header Rules

RX000R: FrscdL1, FaftpL1

1. Check for Record Existence

- **Logic:** The program checks if a record exists in the rsclD1 file based on the STSKEY key.
- **File:** FrscdL1 (File containing resource details)
- **Field:** STSKEY
- **Condition:** The process will not proceed if a record is not found in FrscdL1.

2. Retrieve Firm Name

- **Logic:** If the record is found, the program retrieves the firm name from the rcftpa field.
- **File:** FrscdL1 (File containing resource details)
- **Field:** rcftpa
- **Condition:** This action is contingent upon finding a valid record in FrscdL1.

Configuration and Authorization Rules

1. Check for FTP Parameters

- **Logic:** The program checks for the existence of a record in the aftpl1 file using the aftpl1_key. If not found or if the aflocd field is empty, an error message is generated.

• **Files:**

- FaftpL1 (File containing FTP parameters)

• **Fields:**

- aftpl1_firm (Firm identifier from aftpl1)

- aftpl1_navn (Name from aftpl1)

- **Condition:** The process will generate an error if the record is not found or if aflocd is blank.

2. Error Handling for Missing Local Path

- **Logic:** If the local path (aflocd) is missing, an error message is set to indicate that the local path is required.

- **File:** FaftpL1 (File containing FTP parameters)

- **Field:** aflocd

- **Condition:** This rule is triggered when aflocd is empty.

Financial and Transactional Rules

1. Set Parameters for File Processing

- **Logic:** If valid records are found, the program sets the parameters for file processing, including the path, file name, and job name.
- **File:** FaftpL1 (File containing FTP parameters)
- **Fields:**
 - p_bane (Path for the FTP job)
 - p_file (File name for the FTP job)
- **Condition:** This action occurs only if valid records are found in both FrscdL1 and aftpl1.

2. Program Termination

- **Logic:** The program sets the last record indicator (*inlr) to on, indicating the end of processing.
- **File:** Not applicable
- **Condition:** This occurs at the end of the program execution.

Special Conditions (Program-Specific)

1. Subroutine Initialization (RX000R)

- **Logic:** The program initializes parameters upon entry, setting up the keys for file access.
- **File:** Not applicable
- **Field:** Not applicable
- **Condition:** This is executed when the program is called, preparing it for processing.

2. Key List Definition (RX000R)

- **Logic:** The program defines key lists for accessing records in FrscdL1 and aftpl1.
- **File:** Not applicable
- **Fields:** STSFIR, aftpl1_firm, aftpl1_navn
- **Condition:** This setup is essential for ensuring that the correct records are accessed during processing.

Subprogram Calls Affecting Logic

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

1. AA005R (Error Handling Program)

- **Trigger:** Called when the local path is missing during the FTP parameter check.
- **Logic:** This program handles the error by displaying a message indicating the missing local path.
- **Impact:** This call acts as a **major logical gateway** for error management in the file retrieval process.

2. *INZSR (Initialization Subroutine)

- **Trigger:** Automatically called at the start of the program.
- **Logic:** Initializes parameters and sets up key lists for file access.
- **Impact:** This is a **critical setup step** ensuring that the program has the necessary context for execution.

3. CHAIN Operations (File Access)

- **Trigger:** Executed to retrieve records from FrscdL1 and aftpl1.

- Logic:** These operations check for the existence of records based on defined keys.
- Impact:** These operations are essential for determining the flow of the program and ensuring that the necessary data is available for processing.