

# RX000R.MBR

**Path:** NXCLOUD/rpgsrc/RX000R.MBR **Generated:** 2026-01-08 13:32:59 **Processing Time:** 11218ms

## 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 how the program fetches and validates data related to the FTP parameters.

The core logic for fetching file and path information is contained within the main program logic of RX000R. The program processes records from the rscdL1 and aftpl1 files to determine the appropriate FTP parameters based on the provided firm code.

## Order Status and Header Rules

RX000R: rscdL1, aftpl1

### 1. Retrieve Firm Information

• **Logic:** The program retrieves the firm information based on the provided firm code (p\_firm). If the firm is found in the rscdL1 file, it assigns the firm name to aftpl1\_firm and the corresponding name to aftpl1\_navn.

• **File:** rscdL1 (Firm code register)

• **Field:** STSFIR

• **Condition:** The process will not proceed to fetch the FTP parameters if the firm is not found in rscdL1.

### 2. Check for Local Path

• **Logic:** After retrieving the firm information, the program checks if the local path (aflocd) is empty. If it is, an error message is generated indicating that the local path is missing.

• **File:** aftpl1 (FTP parameters)

• **Field:** aftpl1\_firm, aftpl1\_navn

• **Condition:** The process will generate an error if aflocd is empty or if the firm is not found in aftpl1

## Configuration and Authorization Rules

### 1. Parameter Initialization

• **Logic:** The program initializes parameters for file, path, firm, and job based on the entry parameters passed to it.

• **Files:**

• rscdL1 (Firm code register)

• aftpl1 (FTP parameters)

• **Fields:**

• p\_file (File name)

• p\_bane (Path)

• **Condition:** The parameters must be passed correctly to initialize the program.

### 2. Error Handling for Missing Local Path

- **Logic:** If the local path is missing, the program calls another program (AA005R) to handle the error and pass the error message.
- **File:** aftpl1 (FTP parameters)
- **Field:** aflocd
- **Condition:** The program will call the error handling routine if aflocd is blank.

## Financial and Transactional Rules

### 1. File and Path Assignment

- **Logic:** If the local path is found, the program assigns the values of afflin (file name) and rcjnav (job name) to the respective parameters.
- **File:** aftpl1 (FTP parameters)
- **Fields:**
  - p\_bane (Path)
  - p\_file (File name)
- **Condition:** The assignment occurs only if the local path is valid and found.

### 2. End of Processing

- **Logic:** The program sets the last record indicator (\*inlr) to on, indicating the end of processing.
- **File:** None (Finalization step)
- **Condition:** This step is executed at the end of the program.

## Special Conditions (Program-Specific)

### 1. Initialization Subroutine (RX000R)

- **Logic:** The initialization subroutine sets up the parameters for the program based on the entry list.
- **File:** None (Subroutine setup)
- **Field:** None
- **Condition:** This subroutine is called at the beginning of the program to prepare for processing.

### 2. Key List Definition (RX000R)

- **Logic:** The program defines key lists for accessing the rscdL1 and aftpl1 files based on the firm and name fields.
- **File:** rscdL1, aftpl1
- **Fields:** STSFIR, aftpl1\_firm, aftpl1\_navn
- **Condition:** The keys are defined to facilitate record access for the respective files.

## 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:** This program is called when the local path is missing.
- **Logic:** It 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 FTP job setup.