RX000R.MBR

Path: NXCLOUD/rpgsrc/RX000R.MBR **Generated**: 2026-01-08 12:35:23 **Processing Time**: 10236ms

Business Logic for RX000R

This document outlines the business rules that govern the retrieval of file and path information for FTP jobs, based on an analysis of the RPG program RX000R. The primary focus is on how the program processes records to obtain necessary parameters for FTP operations.

The core logic for retrieving file and path information is contained within the main processing logic of the program. The program checks for the existence of specific records and assigns values to parameters used in FTP jobs.

Order Status and Header Rules

RX000R: rscdL1, aftpl1

1. Record Existence Check

•**Logic:** The program first checks if a record exists in the rscdL1 file based on the STSKEY key. If found, it retrieves the firm and name associated with the record.

•**File:** rscdL1 (Resource Code Directory)

•**Field:** STSFIR

•**Condition:** The process will not proceed if the record is not found in rscdL1.

2. FTP Parameter Validation

•**Logic:** After retrieving the firm information, the program checks for the existence of a corresponding record in the aftpl1 file. If the record is not found or if the local path (aflocd) is empty, an error message is generated.

•**File:** aftpl1 (FTP Parameter List)

•**Field:** aftpl1_firm, aftpl1_navn

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

Configuration and Authorization Rules

1. Parameter Assignment

•**Logic:** If the record is found in aftpl1, the program assigns the local path, file name, and job name to the respective parameters for the FTP job.

•**Files:**

•aftpl1 (FTP Parameter List)

•**Fields:**

•p_bane (Local Path)

•p_file (File Name)

•p_job (Job Name)

•**Condition:** This assignment occurs only if the record in aftpl1 is found and aflocd is not empty.

2. Error Handling

•**Logic:** If the local path is missing, an error message is generated and a specific subroutine (AA005R) is called to handle the error.

•**File:** N/A

•**Field:** w_m50 (Error Message)

•**Condition:** The error handling is triggered when the local path is not found in the FTP parameters.

–

Special Conditions (Program-Specific)

1. Initialization Subroutine

•**Logic:** The program initializes parameters for the FTP job by defining the entry parameters and setting up key fields for file access.

•**File:** N/A

•**Field:** p_file, p_bane, p_firm, p_job

•**Condition:** This initialization occurs at the beginning of the program execution.

2. Key Definitions

•**Logic:** The program defines keys for accessing the rscdL1 and aftpl1 files, ensuring that the correct records are accessed based on the provided parameters.

•**File:** N/A

•**Fields:** STSKEY, aftpl1_key

•**Condition:** Keys are defined at the start of the program to facilitate record access.

–

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 Subroutine)

•**Trigger:** This program is called when the local path is missing in the FTP parameters.

•**Logic:** It receives an error message as a parameter to handle the situation appropriately.

•**Impact:** This call acts as a **major logical gateway** for error management in the FTP job setup.

2. N/A (Initialization Logic)

•**Trigger:** The initialization logic is executed at the start of the program.

•**Logic:** Sets up the necessary parameters and keys for file access.

•**Impact:** This is a **critical setup step** for ensuring the program runs smoothly.

3. N/A (Record Access Logic)

•**Trigger:** The program accesses records based on defined keys.

•**Logic:** It checks for the existence of records in rscdL1 and aftpl1.

•**Impact:** This represents the foundational logic for retrieving necessary data for the FTP job.