IT001R.MBR

Path: NXCLOUD/rpgsrc/IT001R.MBR **Generated**: 2026-01-08 13:32:20 **Processing Time**: 11460ms

Business Logic for IT001R

This document outlines the business rules that govern the process of adding replication transactions to the NAV queue, based on an analysis of the RPG program IT001R. The primary focus is on how parameters are processed and written to a specific file.

The core logic for processing replication transactions is contained within the main program logic of IT001R. The program reads input parameters and writes them to a designated file.

Parameter Handling and File Writing Rules

IT001R: itnast

1. Parameter Assignment

- **Logic:** The program assigns input parameters to internal variables for further processing.
- **File:** itnast (File for replication transactions)
- **Field:** itfirm, itprog, itcomd, itparm, itousr
- **Condition:** The program will assign values to these fields based on the parameters passed from the calling program.

2. File Write Operation

- **Logic:** After assigning the parameters, the program writes the data to the itnast file.
- **File:** itnast (File for replication transactions)
- **Field:** itnastr
- **Condition:** The write operation occurs unconditionally after parameter assignment.

Program Initialization Rules

1. Program Initialization

- **Logic:** The program initializes by reading parameters from the calling program and setting the current date and time.
- **Files:**
- itnast (File for replication transactions)
- **Fields:**
- p_firm (from parameter list)
- p_prog (from parameter list)
- p_comd (from parameter list)
- p_parm (from parameter list)
- p_rpgp (from parameter list)
- **Condition:** This initialization occurs at the start of the program when it is called.

2. Date and Time Capture

- **Logic:** The program captures the current date and time for logging purposes.
- **File:** N/A
- **Field:** w_date, w_time

•**Condition:** This capture occurs unconditionally at the beginning of the program.

Program Termination Rules

1. Program Termination

•**Logic:** The program sets the last record indicator to on and returns control to the calling program.

•**File:** N/A

•**Condition:** This termination occurs unconditionally at the end of the program.

Subprogram Calls Affecting Logic

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

1. *inzsr (Initialization Subroutine)

•**Trigger:** This subroutine is called at the beginning of the program to initialize parameters.

•**Logic:** It captures the input parameters from the calling program and initializes local variables for date and time.

•**Impact:** This call ensures that all necessary parameters are set before the main logic is executed.

2. *entry (Entry Point)

•**Trigger:** This is the entry point for the program where parameters are passed.

•**Logic:** It defines the parameters that the program will accept from the previous program.

•**Impact:** This establishes the context for the program's execution by defining how it interacts with other programs.

3. *return (Return Control)

•**Trigger:** This is invoked at the end of the program.

•**Logic:** It signifies the end of processing and returns control to the calling program.

•**Impact:** This is crucial for maintaining the flow of execution within the overall system, ensuring that control is properly returned after processing.