AN060R.MBR

Path: NXCLOUD/rpgsrc/AN060R.MBR **Generated**: 2026-01-08 12:06:03 **Processing Time**: 11027ms

Business Logic for AN060R

This document outlines the business rules that govern the processing of historical data during a night job, based on an analysis of the RPG program AN060R. The primary focus is on how the program manages subfile interactions and updates historical records.

The core logic for handling historical data is contained within the *inzsr subroutine in AN060R. The program processes user inputs and manages the display and manipulation of records related to night job history.

Order Status and Header Rules

AN060R: ahstpfr

1. Display Subfile Records

•**Logic:** The program reads and displays records from the subfile based on user actions.

•**File:** ahstpfr (Historical Data File)

•**Field:** ahdato

•**Condition:** Records are displayed if the subfile contains entries (srrn01 <> 0).

2. Clear Subfile

•**Logic:** The subfile is cleared of previous entries before new data is populated.

•**File:** b2ctl (Control Record for Subfile)

•**Field:** srrn01

•**Condition:** The subfile is cleared when the program initializes or when a new data set is loaded.

Configuration and Authorization Rules

1. Update Historical Record

•**Logic:** The program updates the historical record if a night job fails, marking it as 'OK'.

•**Files:**

•ahstpfr (Historical Data File)

•ahstlur (Historical Update Record)

•**Fields:**

•ahstl1_dato (Date of Historical Record)

•ahstl1_ttid (Time ID)

•**Condition:** The update occurs when the user selects the option to acknowledge a failed night job (b1valg = '7').

2. Validate Input Data

•**Logic:** The program checks if the input data is valid before processing.

•**File:** b1sfl (Subfile for User Input)

•**Field:** b1dato

•**Condition:** The program will not proceed if b1dato is zero, indicating no valid date is provided.

Financial and Transactional Rules

1. Delete Historical Record

•**Logic:** The program allows the user to delete a historical record based on their selection.

•**File:** ahstlur (Historical Update Record)

•**Fields:**

•ahstl1_dato (Date of Historical Record)

•ahstl1_ttid (Time ID)

•**Condition:** The deletion occurs when the user selects the delete option (b1valg = '4').

2. Chain to Historical Record

•**Logic:** The program chains to a specific historical record to retrieve or update data.

•**File:** ahstpfr (Historical Data File)

•**Condition:** The chaining operation is performed based on the user's selection and the existence of the record.

Special Conditions (Program-Specific)

1. Initialize Program (AN060R)

•**Logic:** The program initializes variables and sets up the necessary conditions for processing.

•**File:** ahstpfr (Historical Data File)

•**Field:** ahstl1_key

•**Condition:** The initialization occurs at the start of the program, resetting counters and preparing for data input.

2. Display Delete Confirmation (AN060R)

•**Logic:** The program displays a confirmation window when the user opts to delete a record.

•**File:** d1win (Delete Confirmation Window)

•**Fields:** d1dato (Date), wactli (Actual Line), wactpo (Actual Position)

•**Condition:** The confirmation window is displayed when the user selects the delete option.

Subprogram Calls Affecting Logic

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

1. xdsp01 (Display Subfile)

•**Trigger:** Called to display the contents of the subfile after data is loaded.

•**Logic:** This subroutine formats and presents the data to the user.

•**Impact:** This call ensures that the user sees the most current data available in the subfile.

2. xclr01 (Clear Subfile)

•**Trigger:** Called to clear the subfile before new data is populated.

•**Logic:** This subroutine resets the subfile indicators and prepares it for new entries.

•**Impact:** This is a critical step to avoid displaying stale data to the user.

3. xd1win (Delete Confirmation Window)

•**Trigger:** Called when the user selects to delete a record.

•**Logic:** This subroutine manages the display and processing of the delete confirmation.

•**Impact:** This represents a significant user interaction point, ensuring that deletions are intentional and confirmed.