

AN060R.MBR

Path: NXCLOUD/rpgsrc/AN060R.MBR **Generated:** 2026-01-09 10:13:44 **Processing Time:** 11948ms

Business Logic for AN060R

This document outlines the business rules that govern the historical query process for night jobs, based on an analysis of the RPG program AN060R. The primary focus is on the logic for managing and displaying historical records related to night jobs.

The core logic for querying historical records is contained within the xdsp01 subroutine in AN060R. This subroutine processes user input and manages the display of records in a subfile format.

Order Status and Header Rules

AN060R: ahstl1, ahstlu, fan060d

1. Display Historical Records

- Logic:** The program retrieves and displays historical records related to night jobs based on user input.
- File:** ahstl1 (Historical records file)
- Field:** ahdato
- Condition:** The process will not display records if the date field ahdato is not valid or if no records are found.

2. Update Historical Record Status

- Logic:** If a night job has failed, the program updates the historical record to reflect this status.
- File:** ahstlu (Historical update file)
- Field:** ahttid
- Condition:** The update occurs when the user selects the option to indicate failure (b1valg = '7').

Configuration and Authorization Rules

1. Subfile Initialization

- Logic:** Initializes the subfile for displaying records, setting necessary counters and indicators.
- Files:**
 - b1sfl (Subfile for displaying records)
 - b2ctl (Control record for subfile)
- Fields:**
 - srrn01 (Current record number)
 - spge01 (Page size for subfile)
- Condition:** This initialization occurs at the start of the program to prepare for user interaction.

2. Clear Subfile

- Logic:** Clears the contents of the subfile before repopulating it with new data.
- File:** b1sfl (Subfile for displaying records)
- Field:** b1snav
- Condition:** This action is triggered whenever a new query is made or the user requests a refresh.

Financial and Transactional Rules

1. Delete Historical Record

- Logic:** The program allows users to delete a selected historical record from the system.
- File:** ahstlu (Historical update file)
- Fields:**
 - ahstl1_dato (Date of the record)
 - ahstl1_ttid (Transaction ID)
- Condition:** The deletion is executed when the user selects the delete option (b1valg = '4').

2. Notify of Invalid Account

- Logic:** The program checks if the user has selected an invalid account and notifies them.
 - File:** b1sfl (Subfile for displaying records)
 - Condition:** This notification occurs if the user input for b2dato is zero.
-

Special Conditions (Program-Specific)

1. Handle Night Job Failure (AN060R)

- Logic:** If a night job fails, the program updates the historical record to mark it as failed.
- File:** ahstlu (Historical update file)
- Field:** ahttid
- Condition:** This check is executed when the user selects the failure option (b1valg = '7'). Note: This logic was added in version 6.10.

2. Display Subfile Records (AN060R)

- Logic:** The program displays the records in the subfile format based on the current record number.
 - File:** b1sfl (Subfile for displaying records)
 - Fields:** b1dato (Date), b1txt1 (Text), b1ttid (Transaction ID)
 - Condition:** This display occurs when there are records to show, indicated by srn01 <> 0.
-

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 Records)

- Trigger:** Called to display the contents of the subfile.
- Logic:** This subroutine manages the display logic and updates the screen based on the current state of the records.
- Impact:** This call acts as a **major logical gateway** for displaying historical records.

2. xd1win (Handle Delete Window)

- Trigger:** Invoked when the user selects to delete a record.
- Logic:** This subroutine manages the confirmation and execution of record deletion.
- Impact:** This is a **critical step** in managing historical records, ensuring user confirmation before deletion.

3. xdatfs (Convert Screen to File Format)

- Trigger:** Called to convert date formats between screen and file representations.
- Logic:** This subroutine handles the conversion of date fields for proper storage and display.
- Impact:** This ensures data integrity when interacting with date fields across different formats.