

RM100R.MBR

Path: NXCLOUD/rpgsrc/RM100R.MBR **Generated:** 2026-01-08 13:32:53 **Processing Time:** 10842ms

Business Logic for RM100R

This document outlines the business rules that govern the maintenance of common property records, based on an analysis of the RPG program RM100R. The primary focus is on the logic for handling subfile operations and user interactions within the program.

The core logic for record maintenance is contained within the various subroutines in RM100R. The program processes user inputs, manages subfile displays, and performs CRUD (Create, Read, Update, Delete) operations on common property records.

Order Status and Header Rules

RM100R: ralmpfr, apospfr

1. Record Selection

- **Logic:** The program selects records based on user input for common property names or identifiers.
- **File:** ralmpfr (Common property records)
- **Field:** ralml1_alme, ralml2_navn
- **Condition:** The process will not select a record if both b2alme and b2navn are zero or blank.

2. Subfile Initialization

- **Logic:** Initializes the subfile for displaying records and sets the necessary indicators.
- **File:** b1sfl (Subfile for common properties)
- **Field:** w_srrn, w_spge
- **Condition:** The subfile is cleared and filled based on the current record positioning.

Configuration and Authorization Rules

1. User Authorization

- **Logic:** Checks user permissions based on the logged-in user's firm and role.
- **Files:**
 - ralmpfr (Common property records)
 - apospfr (Postal codes)
- **Fields:**
 - l_firm (User's firm)
 - w_firm (Current firm context)
- **Condition:** The program will only allow access to records that match the user's firm.

2. Field Validation

- **Logic:** Validates user input fields before processing any updates or inserts.
- **File:** ralmpfr (Common property records)
- **Field:** c2navn, c2ponr
- **Condition:** The process will not proceed if c2navn is blank or if c2ponr does not exist in the postal code file.

Financial and Transactional Rules

1. Record Update Logic

• **Logic:** Updates the common property record with new values if the record exists; otherwise, it creates a new record.

• **File:** ralmpfr (Common property records)

• **Fields:**

• rmnavn (Name of the common property)

• rmadr1 (Address line 1)

• **Condition:** The update occurs if the record is found; otherwise, a new record is created.

2. Record Deletion

• **Logic:** Deletes a selected common property record after user confirmation.

• **File:** ralmpfr (Common property records)

• **Condition:** The deletion is confirmed through a specific user interface prompt.

Special Conditions (Program-Specific)

1. Subfile Display Logic (RM100R)

• **Logic:** Manages the display of records in a subfile format, allowing users to navigate through records.

• **File:** b1sfl (Subfile for common properties)

• **Field:** w_srrn

• **Condition:** The subfile is populated based on the current record positioning and user selections.

Note: This includes handling page navigation.

2. Error Handling and Messaging (RM100R)

• **Logic:** Displays error messages if a user attempts to create a duplicate record.

• **File:** c1msg (Message display)

• **Fields:** b_anul (Cancel flag)

• **Condition:** An error message is shown if the user tries to create a record that already exists.

Subprogram Calls Affecting Logic

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

1. RM105R (Property Details Maintenance)

• **Trigger:** Called when the user selects to view or edit property details.

• **Logic:** Manages the details of a specific common property.

• **Impact:** This call allows for detailed management of property information.

2. RM110R (Property Linking)

• **Trigger:** Invoked when the user chooses to link properties.

• **Logic:** Handles the linking of common properties to other entities.

• **Impact:** This enables the establishment of relationships between records.

3. VB100R (Item Linking)

• **Trigger:** Called when the user selects to link items to the common property.

- Logic:** Manages the association of items with common properties.

- Impact:** This represents the handoff to the next major business function of item management related to properties.