

RM100R.MBR

Path: NXCLOUD/rpgsrc/RM100R.MBR **Generated:** 2026-01-08 15:04:20 **Processing Time:** 15505ms

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 related to subfile handling and user interactions within the program.

The core logic for managing the common property records is contained within the various subroutines in RM100R. The program utilizes subfiles to display records and allows users to perform actions such as create, update, delete, and view details of common properties.

Order Status and Header Rules

RM100R: ralml1, ralml2, ralmlr, ralmlu, apospf

1. Subfile Initialization

- Logic:** Initializes the subfile structure and prepares it for data entry by setting keys and clearing previous data.
- File:** ralml1, ralml2, ralmlr, ralmlu (Common property records)
- Field:** ralml1_alme, ralml2_navn
- Condition:** The subfile is populated when the program starts, and records are fetched based on firm and property identifiers.

2. Record Positioning

- Logic:** Positions the cursor in the subfile based on user input for common property or name.
- File:** ralml1, ralml2 (Common property records)
- Field:** b2alme, b2navn
- Condition:** The positioning occurs if either b2alme or b2navn is not zero or blank.

Configuration and Authorization Rules

1. Subfile Display Control

- Logic:** Controls the visibility of subfile records based on the current record number.
- Files:**
- b1sfl (Subfile for common properties)
- Fields:**
- w_srrn (Relative record number)
- Condition:** The subfile is displayed only if there are records to show (w_srrn is not zero).

2. Error Handling for Existing Records

- Logic:** Displays a message if a user attempts to create a record that already exists.
- File:** ralmlr (Existing common property records)
- Field:** c1alme
- Condition:** This check is performed when the user tries to create a new record, and if the record exists, a message is shown.

Financial and Transactional Rules

1. Record Update Logic

- Logic:** Updates the existing record in the database or creates a new record if it does not exist.
- File:** ralmlu (Updated common property records)
- Fields:**
 - rmnavn (Property name)
 - rmadr1, rmadr2 (Property address)
- Condition:** The record is updated if found; otherwise, a new record is created.

2. Record Deletion Logic

- Logic:** Deletes a selected record from the database.
 - File:** ralmlu (Common property records)
 - Condition:** The deletion occurs when the user confirms the action through the delete window.
-

Special Conditions (Program-Specific)

1. Subfile Refresh Logic (RM100R)

- Logic:** Refreshes the subfile contents based on user actions or navigation.
- File:** b1sfl (Subfile for common properties)
- Field:** b1valg
- Condition:** The subfile is refreshed when the user navigates through the records or performs an action that modifies the data.

2. Copying Logic (RM100R)

- Logic:** Allows the user to copy an existing record to create a new one.
 - File:** ralmlr (Source record for copying)
 - Fields:** k1alme (Key for the record to copy)
 - Condition:** The copy action is triggered when the user selects the copy option from the menu.
-

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:** Handles the maintenance of property details and ensures data integrity.
- Impact:** This call acts as a **major logical gateway** for managing property information.

2. RM110R (Property Linking)

- Trigger:** Called when the user selects to link properties.
- Logic:** Manages the linking of properties to other entities within the system.
- Impact:** This is a **critical step** in ensuring that properties are correctly associated with other records.

3. VB100R (Item Linking)

- Trigger:** Called when the user selects to link items to a property.
- Logic:** Facilitates the association of items with common properties.

- Impact:** This represents the handoff to the next major business function, ensuring that inventory items are correctly linked to properties.