

# RA001R.MBR

**Path:** NXCLOUD/rpgsrc/RA001R.MBR **Generated:** 2026-01-08 15:04:11 **Processing Time:** 18133ms

## Business Logic for Vedlikehold av Statuskoder

This document outlines the business rules that govern the maintenance of status codes 1 and 3, based on an analysis of the RPG program RA001R. The primary focus is on how status codes are retrieved, validated, and updated within the system.

The core logic for maintaining status codes is contained within the A2BLD subroutine in RA001R. The program retrieves existing status codes from the database, performs validation checks, and updates the records accordingly.

## Order Status and Header Rules

Vedlikehold av Statuskoder: raa1l1, raa3l1

### 1. Retrieve Status Code 1

- Logic:** The program retrieves the status code 1 from the raa1l1 file. If the record is found, it populates various fields with the corresponding values from the database.
- File:** raa1l1 (Status Code 1 File)
- Field:** ra1ape, ra1per, ra1aar, ra1hkt, ra1hrs, ra1ars, ra1avs, ra1bnr, ra1bun, ra1skd, ra1pre
- Condition:** The record is only processed if the chain operation is successful (\*in60 is set to \*on).

### 2. Retrieve Status Code 3

- Logic:** The program retrieves the status code 3 from the raa3l1 file. Similar to status code 1, it populates fields with values from the database if the record is found.
- File:** raa3l1 (Status Code 3 File)
- Field:** ra3stp, ra3p01, ra3p02, ra3p03, ra3p04, ra3p05, ra3p06, ra3p07, ra3p08, ra3p09, ra3p10, ra3p11, ra3p12, ra3p13
- Condition:** The record is only processed if the chain operation is successful (\*in61 is set to \*on).

## Configuration and Authorization Rules

### 1. Validation of Periods

- Logic:** The program checks that the period (a21ape) is within valid bounds (1 to 13). If not, it sets an error indicator.
- Files:**
  - raa1l1 (Status Code 1 File)
  - raa3l1 (Status Code 3 File)
- Fields:**
  - a21ape (Current Period from Status Code 1)
- Condition:** The process will not proceed if a21ape is less than 1 or greater than 13.

### 2. Year Validation

- Logic:** The program checks that the year (a21aar) is greater than or equal to 1980 if the period (a21per) is greater than zero. If not, it sets an error indicator.
- File:** raa1l1 (Status Code 1 File)

- Field:** a21aar

- Condition:** The process will not proceed if a21per is greater than zero and a21aar is less than 1980.

–

## Financial and Transactional Rules

### 1. Check for Period Lock

- Logic:** The program checks if the current period is locked (a21per > a21ape). If it is, an error indicator is set.

- File:** raa1l1 (Status Code 1 File)

- Field:** a21per

- Condition:** The process will not proceed if a21per is greater than a21ape.

### 2. Validate Accounting Year for Status Code 3

- Logic:** The program checks if the accounting year for status code 3 is valid. If any of the periods (a23p01 to a23p13) are zero or exceed the current period, an error indicator is set.

- File:** raa3l1 (Status Code 3 File)

- Fields:** a23p01, a23p02, a23p03, a23p04, a23p05, a23p06, a23p07, a23p08, a23p09, a23p10, a23p11, a23p12, a23p13

- Condition:** The process will not proceed if any of the periods are zero or exceed a21ape.

–

## Special Conditions (Program-Specific)

### 1. Update Status Code 1 (RA001R)

- Logic:** After validation, the program updates the status code 1 record with the new values. If the record is not found, it writes a new record.

- File:** raa1lu (Status Code 1 Update File)

- Field:** ra1ape, ra1per, ra1aar, ra1hkt, ra1hrs, ra1ars, ra1avs, ra1bnr, ra1bun, ra1skd, ra1pre

- Condition:** The update occurs if the chain operation is successful (\*in60 is set to \*off).

### 2. Update Status Code 3 (RA001R)

- Logic:** Similar to status code 1, the program updates the status code 3 record with the new values or writes a new record if not found.

- File:** raa3lu (Status Code 3 Update File)

- Field:** ra3stp, ra3p01, ra3p02, ra3p03, ra3p04, ra3p05, ra3p06, ra3p07, ra3p08, ra3p09, ra3p10, ra3p11, ra3p12, ra3p13

- Condition:** The update occurs if the chain operation is successful (\*in61 is set to \*off).

–

## Subprogram Calls Affecting Logic

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

### 1. A2BLD (Status Code Maintenance Screen)

- Trigger:** Called to display the maintenance screen for status codes 1 and 3.

- Logic:** Displays the current values of the status codes and allows for user input to update them.

- Impact:** This call acts as a **major logical gateway** for user interaction and validation.

### 2. keya1 and keya3 (Key Definitions)

- Trigger:** Used to define keys for accessing status code records.
- Logic:** Sets up keys based on the firm number for both status codes.
- Impact:** Essential for ensuring that the correct records are accessed and modified.

### **3. \*inzsr (Initialization Subroutine)**

- Trigger:** Called at the beginning of the program to initialize variables.
- Logic:** Moves the firm number and name into working variables.
- Impact:** This ensures that the program operates with the correct context for the firm being processed.