

# VL001R.MBR

**Path:** NXCLOUD/rpgsrc/VL001R.MBR **Generated:** 2026-01-08 12:35:39 **Processing Time:** 15002ms

## Business Logic for Inventory Update Process

This document outlines the business rules that govern the inventory update process, based on an analysis of the RPG program VL001R. The primary focus is on how inventory records are updated and logged in the historical inventory book.

The core logic for updating inventory is contained within the main processing logic of the VL001R program. The program handles various conditions for updating inventory based on transaction types, item codes, and inventory locations.

## Inventory Status and Update Rules

**Inventory Update Process:** vlaglu, vhislu, vvarl1, vugrl1, vltyl1, votyl1, vvenl1, ra10l1, rmpil1, rmpilu

### 1. Valid Item Check

- Logic:** The program checks if the item exists in the inventory register. If not found, the process is aborted, and an error is logged.
- File:** vvarl1 (Item register)
- Field:** vvarl1\_vare
- Condition:** The process will not continue if the item is not found (\*in90 is set).

### 2. Valid Location Check

- Logic:** The program verifies if the specified inventory location is valid. If the location is invalid, an error is logged, and the process is aborted.
- File:** ra10l1 (Location register)
- Field:** ra10l1\_jkod
- Condition:** The process will not continue if the location is not found (\*in90 is set).

### 3. Transaction Date and Time Initialization

- Logic:** If the transaction date or time is not provided, the system date and time are assigned.
- File:** N/A
- Field:** hldato, hltime
- Condition:** The date and time are set to the current system date and time if they are in their initial state (\*loval).

### 4. Order Type and Alternative Processing Code Check

- Logic:** The program checks if both the order type and alternative processing code are blank. If both are blank, the process is aborted.
- File:** N/A
- Field:** hlotyp, hlaltk
- Condition:** The process will not continue if both fields are blank.

## Configuration and Authorization Rules

### 1. Inventory Type Validation

- Logic:** The program validates the inventory type against the order type. If the inventory type does not match, an error is logged.
- File:** vltyl1 (Line type register)
- Field:** vltyl1\_ltyp
- Condition:** The process will not continue if the inventory type is not valid.

## 2. Under Group Inventory Control Check

- Logic:** The program checks if the under group is set to be inventory controlled. If it is not, an error is logged.
- File:** vugrl1 (Under group register)
- Field:** vgulag
- Condition:** The process will not continue if vgulag is set to 1.

# Financial and Transactional Rules

## 1. Transaction Type Handling

- Logic:** The program handles different transaction types (e.g., incoming, outgoing, adjustments) and updates inventory accordingly.
- File:** vhislu (Historical inventory book)
- Fields:**
  - vhislu\_lage (Location)
  - vhislu\_vare (Item)
- Condition:** The process updates inventory based on the specified transaction type.

## 2. Adjustment Handling

- Logic:** If an adjustment transaction is detected, the program calculates the difference and updates the inventory accordingly.
- File:** vhislu (Historical inventory book)
- Condition:** The process will adjust inventory levels based on the adjustment type.

# Special Conditions (Program-Specific)

## 1. MalProff Update Handling (VL001R)

- Logic:** If the transaction is related to MalProff, additional updates to the supplementary register are performed.
- File:** rmpil1, rmpilu (Supplementary register)
- Field:** rmpil1\_var
- Condition:** The process checks if the transaction type is 'MP' and updates the supplementary register accordingly.

## 2. Inventory Creation Handling (VL001R)

- Logic:** If the inventory record does not exist, a new record is created.
- File:** vlaglu (Inventory register)
- Field:** vlaglu\_vare
- Condition:** The process will create a new inventory record if it is not found (\*in90 is set).

# Subprogram Calls Affecting Logic

### **1. CO402R (External Program for MalProff)**

- Trigger:** Called during initialization to check if MalProff updates are required.
- Logic:** This program checks parameters related to MalProff and sets a flag if updates are needed.
- Impact:** This affects whether the supplementary register is updated during the inventory process.

### **2. VL710R (Retrieve Item Type)**

- Trigger:** Called to retrieve the item type based on the inventory location.
- Logic:** This program fetches the item type and other related parameters.
- Impact:** This is crucial for determining if the item can be processed as inventory.

### **3. RMP97C (Logging Transactions)**

- Trigger:** Called at the end of the program to log transaction details.
- Logic:** This program logs the transaction information for auditing purposes.
- Impact:** This ensures that all inventory transactions are recorded for future reference.

This documentation serves as a comprehensive guide to the business logic implemented in the VL001R program, detailing the rules and conditions that govern inventory updates within the system.