

# VP110R.MBR

**Path:** NXCLOUD/rpgsrc/VP110R.MBR **Generated:** 2026-01-09 10:13:45 **Processing Time:** 12331ms

## Business Logic for Vare-pris Maintenance

This document outlines the business rules that govern the maintenance of item prices within the ERP system, based on an analysis of the RPG program VP110R. The primary focus is on the logic related to the management of item prices, including the addition, deletion, and display of prices in a subfile format.

The core logic for price maintenance is contained within the init subroutine in VP110R. The program processes records related to item prices, allowing users to view, update, and delete prices associated with various items and suppliers.

## Order Status and Header Rules

Vare-pris Maintenance: vvprpfr, vvprl1r, vvprlur, vvarpfr, vvenpfr, vltpfr, ra30pfr, venhpfr

### 1. Display Subfile Records

- Logic:** The program displays item prices in a subfile format, allowing users to navigate through the records.
- File:** vvprpfr (Item Price File)
- Field:** Various fields related to item prices.
- Condition:** The subfile is populated based on the current item and supplier.

### 2. Delete Price Records

- Logic:** Users can delete price records, but checks are performed to ensure that no orders exist for the item before deletion.
- File:** vvprpfr (Item Price File)
- Field:** b1valg
- Condition:** The deletion is only allowed if b\_best is not set to \*on, indicating that there are no outstanding orders.

## Configuration and Authorization Rules

### 1. Price Group Validation

- Logic:** The program checks if a price group is valid before allowing the user to proceed with actions related to item prices.
- Files:**
  - vvprpfr (Item Price File)
  - ra30l1 (Price Group File)
- Fields:**
  - vvprlr\_prgr (Price Group)
  - ra30l1\_dkod (Price Group Code)
- Condition:** The price group must exist in the ra30l1 file for the operation to proceed.

### 2. Supplier Existence Check

- Logic:** The program checks if the supplier exists before allowing price updates or deletions.

- File:** vvprpfr (Item Price File)
  - Field:** w\_ldor
  - Condition:** The supplier must be valid and exist in the relevant supplier file.
- 

## Financial and Transactional Rules

### 1. Price Calculation Logic

- Logic:** The program calculates the coverage percentage based on the cost price and selling price.
- File:** vvprpfr (Item Price File)
- Fields:**
  - b1kopr (Cost Price)
  - b1sapr (Selling Price)
- Condition:** If the cost price is zero, it retrieves the coverage from the item or subgroup.

### 2. Update Price Records

- Logic:** When updating prices, the program ensures that the new price does not conflict with existing records.
  - File:** vvprpfr (Item Price File)
  - Condition:** The update is only performed if the new price is valid and does not create inconsistencies.
- 

## Special Conditions (Program-Specific)

### 1. Supplier Deletion Logic (VP110R)

- Logic:** If the last price for a supplier is deleted, the supplier is removed from the inventory.
- File:** vvprpfr (Item Price File)
- Field:** w\_ldor
- Condition:** This check ensures that if there are no remaining prices for a supplier, they are removed from the inventory.

### 2. Order Check Before Deletion (VP110R)

- Logic:** The program checks for any existing orders before allowing the deletion of an item price.
  - File:** lodtlv (Order Details File)
  - Fields:** d1enhe (Item Unit)
  - Condition:** If there are orders associated with the item, deletion is blocked.
- 

## Subprogram Calls Affecting Logic

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

### 1. CO402R (Price Group Check)

- Trigger:** Called during the initialization of the program.
- Logic:** This program checks if the price group is blank and sets a flag accordingly.
- Impact:** This affects the visibility of prices based on the price group settings.

### 2. xh1win (Display Handling)

- Trigger:** Called when displaying the item price window.

- Logic:** This program handles the display of the item price interface.
- Impact:** It ensures that the correct data is shown to the user based on their selections.

### **3. sjekk\_best (Order Check)**

- Trigger:** Called when checking if an item is on order.
- Logic:** This program checks the order details to see if any orders exist for the item.
- Impact:** It prevents deletion of prices if orders are present, maintaining data integrity.