

LX401R.MBR

Path: NXCLOUD/rpgsrc/LX401R.MBR **Generated:** 2026-01-08 12:35:01 **Processing Time:** 9452ms

Business Logic for Inventory Unit Transfer to Inventory Register

This document outlines the business rules that govern the transfer of inventory units from the inventory register, based on an analysis of the RPG program LX401R. The primary focus is on the logic that reads inventory records, updates inventory units, and maintains tracking information. The core logic for transferring inventory units is contained within the main program section of LX401R. The program reads records from the inventory register, finds the corresponding inventory unit in the item register, and updates the inventory unit details accordingly.

Inventory and Record Update Rules

Inventory Transfer: vlaglu, vvarl1

1. Read Inventory Register

• **Logic:** The program reads records from the inventory register file (vlaglu) until no more records are available.

• **File:** vlaglu (Inventory Register)

• **Field:** vlvare

• **Condition:** The process continues to read records as long as the end-of-file indicator (*in90) is off.

2. Update Inventory Unit

• **Logic:** For each inventory record read, the program retrieves the associated item from the item register and updates the inventory unit with the current date, time, and user information.

• **File:** vvarl1 (Item Register)

• **Field:** vvarl1_vare

• **Condition:** The update occurs after successfully retrieving the item record.

User and Tracking Information Rules

1. User Tracking Information

• **Logic:** The program captures the user ID, date, and time when the inventory unit is updated, ensuring traceability of changes.

• **Files:**

• vlaglur (Updated Inventory Register)

• **Fields:**

• vleusr (User ID from local data area)

• vledat (Current date)

• vletim (Current time)

• **Condition:** This information is set during the update process for each inventory unit.

Special Conditions (Program-Specific)

1. Program Initialization (LX401R)

•**Logic:** The program initializes by defining keys for lookups in the item register based on the firm and item variables.

•**File:** vvarl1 (Item Register)

•**Field:** w_firm

•**Condition:** The firm variable (l_firm) is set at the beginning of the program to ensure correct context for lookups.

2. Program Termination (LX401R)

•**Logic:** The program sets the last record indicator (*inlr) to on before returning control to the calling program, ensuring proper termination.

•**File:** None (Program Logic)

•**Condition:** This occurs at the end of the program execution.

Subprogram Calls Affecting Logic

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

1. *inzsr (Initialization Subroutine)

•**Trigger:** This subroutine is called at the beginning of the program.

•**Logic:** It initializes the keys for the item register lookup based on the firm and item variables.

•**Impact:** This initialization ensures that the program can accurately retrieve and update item records based on the current firm context.

2. avslutt (Termination Tag)

•**Trigger:** This tag is reached at the end of the program.

•**Logic:** It sets the last record indicator and returns control to the calling program.

•**Impact:** This ensures that the program exits cleanly, maintaining system stability.

This documentation provides a comprehensive overview of the business logic implemented in the LX401R program, focusing on the key processes involved in transferring inventory units and maintaining tracking information.