NO100R.MBR

Path: NXCLOUD/rpgsrc/NO100R.MBR **Generated**: 2026-01-08 13:32:42 **Processing Time**: 17919ms

Business Logic for External Order Header Maintenance

This document outlines the business rules that govern the maintenance of external order headers, based on an analysis of the RPG program NO100R. The primary focus is on the logic that handles the processing and validation of external orders, including their associated lines and statuses.

The core logic for maintaining external order headers is contained within the various subroutines in NO100R, which processes records from multiple files related to external orders, customer orders, and inventory.

Order Status and Header Rules

External Order Maintenance: NOHEL2, NOHEL3, NOHEL4, NOHEL5, NOHEL6

1. Check for Existing Orders

- **Logic:** The program checks if an external order already exists in the NOHEL files before allowing an update or creation.
- **File:** NOHEL2 (External Order Header)
- **Field:** NREF
- **Condition:** The process will not select a record if NREF is not equal to the current reference number being processed.

2. Validation of Order Status

- **Logic:** The order status is validated to ensure it is not in a state that prevents further processing.
- **File:** NOHEL (External Order Header)
- **Field:** STAT
- **Condition:** The process will not proceed if STAT is not equal to 0 or 8.

Configuration and Authorization Rules

1. Customer and Project Creation

- **Logic:** If a customer or project needs to be created, the program calls the appropriate subroutine to handle this.
- **Files:**
- AUSRL1 (User Information)
- FKPRL1 (Customer Project)
- **Fields:**
- USER (from AUSRL1)
- KUND (from FKPRL1)
- **Condition:** The creation process is triggered when the function key for creating a customer or project is pressed.

2. EDI Order Handling

- **Logic:** The program checks if the order is being processed as an EDI order and adjusts the handling accordingly.

- **File:** NOHEL (External Order Header)
- **Field:** FSYS
- **Condition:** The process will treat the order differently if FSYS is equal to 'EDI', 'SMK', or 'EHF'.

Financial and Transactional Rules

1. Order Transfer to Inventory

- **Logic:** The program checks if the order can be transferred to inventory based on existing conditions and statuses.
- **File:** NODTL (Order Lines)
- **Fields:**
- TELL (Order Number)
- FSYS (System Type)
- **Condition:** The transfer will not occur if the order is already marked as transferred or if it contains errors.

2. Inventory Check for Order Lines

- **Logic:** The program verifies if there are any inventory items associated with the order lines before proceeding.
- **File:** NODTL (Order Lines)
- **Condition:** The process will not proceed if there are no valid inventory items linked to the order.

Special Conditions (Program-Specific)

1. Handling of Locked Orders (NO120R)

- **Logic:** The program checks if the order is locked by another process before allowing modifications.
- **File:** NOHEL (External Order Header)
- **Field:** TELL
- **Condition:** The process will not allow updates if the order is locked.

2. Updating Delivery Dates

- **Logic:** When a delivery date is set, all associated lines must also be updated to reflect this change.
- **File:** NODTL (Order Lines)
- **Fields:** LDAT (Delivery Date)
- **Condition:** The update will occur only if the delivery date is valid and not equal to a placeholder value.

Subprogram Calls Affecting Logic

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

1. NO110R (Status Codes Retrieval)

- **Trigger:** Called when the status key is pressed.
- **Logic:** Retrieves and displays the status codes associated with the order.
- **Impact:** This call acts as a **major logical gateway** for managing order statuses.

2. NO150R (Customer Order Retrieval)

- **Trigger:** Called when the function key for retrieving customer orders is pressed.
- **Logic:** Fetches customer orders related to the current external order.
- **Impact:** This is a **destructive filtering step** that limits the orders displayed based on the current context.

3. FA720R (Job Register Update)

- **Trigger:** Called during the process of updating job statuses.
- **Logic:** Updates the job register with the current status of the order processing.
- **Impact:** This call ensures that the job register reflects the latest processing state, which is critical for tracking order handling.

Use clear, business-friendly language. Fill in all sections with details from the code. Output only the Markdown.