

# NO100R.MBR

**Path:** NXCLOUD/rpgsrc/NO100R.MBR **Generated:** 2026-01-08 15:04:09 **Processing Time:** 15948ms

## Business Logic for External Order Header Maintenance

This document outlines the business rules that govern the maintenance of external order headers in the ASOFAK system, based on an analysis of the RPG program NO100R. The primary focus is on the logic that handles the creation, updating, and deletion of external order headers and their associated lines.

The core logic for maintaining external order headers is contained within various subroutines in NO100R. The program processes user input through function keys, manages subfile displays, and interacts with multiple database files to ensure data integrity and business rules compliance.

### Order Status and Header Rules

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

#### 1. Check for Existing Orders

- **Logic:** The program checks if an order already exists in the database before allowing updates or deletions.

- **File:** NOHEL2 (External Order Header)

- **Field:** NOHEL2\_NREF

- **Condition:** The process will not allow updates if the order number is not found in NOHEL2.

#### 2. Prevent Duplicate Entries

- **Logic:** If an order is already marked as transferred, it cannot be transferred again.

- **File:** NOHEL2 (External Order Header)

- **Field:** B1OVER

- **Condition:** The process will not allow a transfer if B1OVER is set to 1.

### Configuration and Authorization Rules

#### 1. EDI Order Handling

- **Logic:** If the order is received through EDI, specific checks are performed to ensure it is processed correctly.

- **Files:**

- NOHEL2 (External Order Header)

- NOHEL1 (External Order Line)

- **Fields:**

- NOHEL2\_FSYS (File System)

- NOHEL1\_TELL (Order Number)

- **Condition:** The process checks if the order is EDI and applies different logic for handling it.

#### 2. User Permissions

- **Logic:** The program verifies user permissions before allowing the creation or modification of orders.

- **File:** AUSR (User Registration)
  - **Field:** AUSR\_USER
  - **Condition:** The process will not allow modifications if the user does not have the necessary permissions.
- 

## Financial and Transactional Rules

### 1. Order Line Updates

- **Logic:** When updating an order, all associated order lines must also be updated to reflect the changes.

• **File:** NODTL (Order Line)

• **Fields:**

• NODTL\_TELL (Order Number)

• NODTL\_FSYS (File System)

- **Condition:** The process will ensure that all lines are updated if the header is modified.

### 2. Status Checks

- **Logic:** The program checks the status of orders before allowing further processing.

• **File:** NOHEL2 (External Order Header)

- **Condition:** The process will not allow further actions if the status is not compliant (e.g., status must be less than 9).
- 

## Special Conditions (Program-Specific)

### 1. Handling of Locked Orders (NO120R)

- **Logic:** The program checks if an order is locked before allowing modifications.

• **File:** NOHEL2 (External Order Header)

• **Field:** NOHEL2\_TELL

- **Condition:** The process will not allow changes if the order is locked.

### 2. Order Splitting Logic

- **Logic:** If not all lines of an order have delivery dates, the order will be split into separate entries.

• **File:** NODTL (Order Line)

• **Fields:** NODTL\_LDAT (Delivery Date)

- **Condition:** The process will split the order if any line lacks a delivery date.
- 

## 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 user requests status codes.

• **Logic:** Retrieves status codes related to orders.

• **Impact:** This call provides essential feedback to users regarding order processing.

### 2. NO150R (Customer Order Retrieval)

- **Trigger:** Invoked when the user requests to retrieve customer orders.

• **Logic:** Fetches customer order details based on the order number.

- Impact:** This is a critical step in ensuring that the correct customer order details are displayed.

### **3. FA930R (Customer Creation)**

- Trigger:** Called when creating a new customer.

- Logic:** Handles the logic for adding a new customer to the database.

- Impact:** This call ensures that new customers are correctly registered in the system.

### **4. ND105R (Order Line Processing)**

- Trigger:** Called during order line updates.

- Logic:** Processes updates to order lines based on the current order header.

- Impact:** This represents a significant part of the order processing workflow, ensuring that line items are correctly managed.

-

This documentation provides a comprehensive overview of the business logic implemented in the NO100R program, detailing the rules and processes that govern external order header maintenance within the ASOFAK system.