

# FO100R.MBR

**Path:** NXCLOUD/rpgsrc/FO100R.MBR **Generated:** 2026-01-08 12:58:10 **Processing Time:** 16314ms

## Business Logic for Order Registration

This document outlines the business rules that govern the registration of offers/orders, based on an analysis of the RPG program FO100R. The primary focus is on the validation and processing logic for order registration, including customer and order type checks, as well as handling various conditions related to order management.

The core logic for order registration is contained within the hente\_ordre subroutine in FO100R. The program processes customer information, validates order types, and manages the registration of orders, ensuring that all necessary conditions are met before an order can be successfully registered.

## Order Status and Header Rules: FO100R

### 1. Order Type Validation

- **Logic:** The program checks if the order type has a system code of 0, which is required for valid sales orders.
- **File:** VOTYL1 (Order Type Register)
- **Field:** VOTYL1\_OTYP
- **Condition:** The process will not select a record if the order type does not meet the system code requirement.

### 2. Customer Credit Limit Check

- **Logic:** If the customer has exceeded their credit limit, the program will prevent further order processing unless the order type is 'T' (offer).
- **File:** RKKUND (Customer Register)
- **Field:** RKKRES (Credit Status)
- **Condition:** The process will block the order registration if the customer has a credit hold (s\_sper = 1) and the order type is not 'T'.

## Configuration and Authorization Rules

### 1. User Permissions for Customer Creation

- **Logic:** The program checks if the user has permission to create customers directly from the order system.
- **Files:**
  - FUSR1 (User Register)
- **Fields:**
  - FUSR1\_USER
- **Condition:** The process will block customer creation if the user does not have the necessary permissions.

### 2. Requisition Requirement Notification

- **Logic:** If the customer requires a requisition for orders, the program will notify the user if the requisition field is blank.

- **File:** RKUNL1 (Customer Register)
  - **Field:** RKREKV (Requisition Requirement)
  - **Condition:** The process will alert the user if the requisition field is empty when the customer requires it.
- 

## Financial and Transactional Rules

### 1. Order Date Validation

- **Logic:** The program checks if the order date is valid and not in the past.
- **File:** FOHEUR (Order Header Register)
- **Condition:** The process will block the registration if the order date is less than the current date.

### 2. Delivery Date Validation

- **Logic:** The program checks if the delivery date is valid and not earlier than the order date.
  - **File:** FOHEUR (Order Header Register)
  - **Condition:** The process will block the registration if the delivery date is earlier than the order date.
- 

## Special Conditions (Program-Specific)

### 1. Credit Limit Warning for Offers (FO100R)

- **Logic:** The program allows offers to be created even if the customer has a credit hold, but requires approval for processing.
- **File:** RKKUND (Customer Register)
- **Field:** RKKRES (Credit Status)
- **Condition:** The process will allow the creation of offers but will require additional validation if the credit limit is exceeded.

### 2. Customer Project Requirement Check (FO100R)

- **Logic:** The program checks if the customer has a project requirement and prompts the user to fill it if necessary.
  - **File:** FKPR1 (Customer Project Register)
  - **Fields:** FKPR1\_KUND, FKPR1\_KPRO
  - **Condition:** The process will block the registration if the customer requires a project and it is not provided.
- 

## Subprogram Calls Affecting Logic

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

### 1. CO402R (Configuration Check Program)

- **Trigger:** Called multiple times throughout the program to check various configuration settings.
- **Logic:** This program checks if certain features are enabled based on configuration settings.
- **Impact:** This call acts as a **major logical gateway** for determining the behavior of the order registration process.

### 2. FO730R (Order Processing Program)

- **Trigger:** Called during the finalization of order registration.

- **Logic:** This program handles the final processing of the order, including updating inventory and generating necessary documentation.
- **Impact:** This is a **destructive filtering step** that ensures all order conditions are met before completion.

### 3. FD100R (Order Line Processing Program)

- **Trigger:** Called for registering or updating order line items.
- **Logic:** This program processes individual line items associated with the order, ensuring they meet all requirements.
- **Impact:** This call ensures that line items are correctly associated with the order and validates their data.

-

This documentation provides an overview of the business logic implemented in the FO100R program, detailing the rules and conditions that govern the registration of offers and orders within the system.