FO100R.MBR

Path: NXCLOUD/rpgsrc/FO100R.MBR **Generated**: 2026-01-08 13:32:11 **Processing Time**: 16871ms

Business Logic for Order Registration

This document outlines the business rules that govern the process of registering offers and orders, based on an analysis of the RPG program FO100R. The primary focus is on the validation and processing logic involved in creating and maintaining orders within the system.

The core logic for order registration is contained within various subroutines in FO100R. The program handles user inputs, validates customer and order details, and updates the database accordingly.

Order Status and Header Rules

Order Registration: FOHEUR, FOHEURR, FOHEURD

1. Order Type Validation

- **Logic:** The program checks if the specified order type has a system code of 0 (Sales/Ofak). If not, the order cannot be processed.
- **File:** FOHEUR (Order Header)
- **Field:** FOOTYP
- **Condition:** "The process will not select a record if FOOTYP does not equal 0."

2. Customer Credit Check

- **Logic:** Orders can be registered even if the customer has a credit hold, but a warning is displayed if the credit limit is exceeded.
- **File:** RKKUND (Customer)
- **Field:** RKSALD
- **Condition:** "If the customer's balance exceeds the credit limit, a warning is issued."

Configuration and Authorization Rules

1. User Permissions for Customer Creation

- **Logic:** Users must have permission to create customers directly from the order system. If not, a message is displayed.
- **Files:**
- AUSR (User)
- RKKUND (Customer)
- **Fields:**
- AUSR_USER (from AUSR)
- RKKUND (from RKKUND)
- **Condition:** "The process will not allow customer creation if the user lacks the necessary permissions."

2. Delivery Type Retrieval

- **Logic:** The program retrieves the delivery type from the department or customer if specified.
- **File:** RKKUND (Customer)

- **Field:** RKLITY
- **Condition:** "If a delivery type is specified, it will be retrieved from the customer record."

Financial and Transactional Rules

1. Order Date Validation

- **Logic:** The order date must be valid and cannot be in the future. If invalid, an error message is displayed.
- **File:** FOHEUR (Order Header)
- **Fields:**
- B1BDAT (Order Date)
- B1DDAT (Follow-up Date)
- **Condition:** "The order date must not be greater than the current date."

2. Customer Credit Limit Check

- **Logic:** If the customer's credit limit is exceeded, the order may still be registered if it is an offer, but a warning is displayed.
- **File:** RKKUND (Customer)
- **Condition:** "If the order type is 'T' (offer), it can be registered even if the credit limit is exceeded."

Special Conditions (Program-Specific)

1. Requisition Requirement Check

- **Logic:** If the customer requires a requisition, the program checks if the requisition field is filled. If not, an error is raised.
- **File:** RKKUND (Customer)
- **Field:** RKREKV
- **Condition:** "If the requisition field is blank for a customer that requires it, an error is raised."

2. Order Number Handling

- **Logic:** If an order number is directly entered, the program checks if it exists and retrieves the associated order details.
- **File:** FOHEUR (Order Header)
- **Fields:** FOHEUR_NUMM, FOHEUR_SUFF
- **Condition:** "If the order number exists, the program will retrieve its details; otherwise, an error is raised."

Subprogram Calls Affecting Logic

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

1. AA005R (User Permission Check)

- **Trigger:** Called when a user attempts to create a customer.
- **Logic:** Checks if the user has permission to create customers.
- **Impact:** If the user lacks permission, a message is displayed, and the process is halted.

2. FO101R (Order Header Maintenance)

- **Trigger:** Called when maintaining order header details.

- **Logic:** Updates order header information based on user input.
- **Impact:** Ensures that the order header is correctly updated in the database.

3. FO730R (Order Line Processing)

- **Trigger:** Called during the processing of order lines.
- **Logic:** Handles the registration and updating of order line details.
- **Impact:** This is a critical step in ensuring that all order line information is accurately recorded.

This documentation serves as a guide for understanding the business logic implemented in the FO100R program, focusing on the rules and conditions that govern the order registration process.