

FO602R.MBR

Path: NXCLOUD/rpgsrc/FO602R.MBR **Generated:** 2026-01-21 15:55:36 **Processing Time:** 22860ms

Business Logic for Order Processing and Selection

This document outlines the business rules that govern the order processing and selection from the order register, based on an analysis of the RPG program FO602R. The primary focus is on the conditions and logic that determine how orders are selected, processed, and validated before invoicing.

The core logic for order selection and processing is contained within various subroutines in FO602R. The program handles multiple order types, validates order details, and manages user permissions and conditions for invoicing.

Order Status and Header Rules

Order Processing: FO602D, Rknl1, Fkprl1, Votyl1, Fohelu, Fohel1, Fohelo, Fplolu, Vinfl1, Fusrl1, Fstsl1, Fntrl2, Fntrl1, Fscnl1, Fodtlr, Sdepl1

1. Order Type Validation

- Logic:** The program checks if the specified order type exists in the order type register before processing.
- File:** Votyl1 (Order Type Register)
- Field:** Votyl1_otyp
- Condition:** The process will not proceed if the order type is not found in the register.

2. Order Date Validation

- Logic:** If a "to" date is specified, it checks if the order date is valid and within the specified range.
- File:** Fohelu (Order Header)
- Field:** Foldat
- Condition:** The process will skip orders if the order date exceeds the specified "to" date.

Configuration and Authorization Rules

1. User Authorization for Invoicing

- Logic:** The program checks if the user has permission to initiate invoicing based on their role.
- Files:**
 - Fusrl1 (User Register)
 - Fstsl1 (Ofak Code Register)
- Fields:**
 - Fusrl1_user (User ID from User Register)
 - Fstsl1_key (Ofak Code)
- Condition:** The process will not allow invoicing if the user lacks the necessary permissions.

2. Deposit Check

- Logic:** The program checks if there is an outstanding deposit requirement for the order.
- File:** Sdepl1 (Deposit File)

- Field:** Sdepl1_kund
 - Condition:** The order will not be processed if there is an unpaid deposit.
-

Financial and Transactional Rules

1. Order Line Check

- Logic:** Before processing an order, the program verifies if there are any order lines associated with it.
- File:** Fodtlr (Order Line Register)
- Fields:**
 - Fodtlr_numm (Order Number)
 - Fodtlr_suff (Order Suffix)
- Condition:** If no order lines exist, the order will be flagged for deletion.

2. Invoice Generation Conditions

- Logic:** The program checks various conditions to determine if an invoice can be generated for the order.
 - File:** Fohel1 (Order Header)
 - Condition:** Invoicing will be blocked if the order type requires special handling or if the order has been previously invoiced.
-

Special Conditions (Program-Specific)

1. Handling Special Order Types (FO602R)

- Logic:** The program includes specific logic for handling special order types such as "TILBUD" (Offers) and "OBEK" (Picking Lists).
- File:** Fohelo (Order Header for Picking)
- Field:** Fohelo_otyp
- Condition:** The program will branch to different processing logic based on the order type.

2. Collective Invoicing Logic (FO602R)

- Logic:** The program allows for collective invoicing of debit and credit simultaneously.
 - File:** Fntrl2 (Transaction Type)
 - Fields:** Fntrl2_kftp (Transaction Type)
 - Condition:** The program checks if the transaction type allows for collective invoicing before proceeding.
-

Subprogram Calls Affecting Logic

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

1. FO603R (Invoice Processing Program)

- Trigger:** Called when collective invoicing is enabled.
- Logic:** This program handles the actual creation of invoices based on the selected orders.
- Impact:** This call acts as a **major logical gateway** for invoice generation.

2. FO410R (Order Deletion Notification)

- Trigger:** Called if an order is found to have no associated order lines.

- Logic:** This program sends a notification regarding the deletion of the order.
- Impact:** Ensures that users are informed of any orders that are deleted due to lack of lines.

3. CO402R (Collective Invoicing Control)

- Trigger:** Called to check if collective invoicing should be enabled.
- Logic:** This program checks the parameters for collective invoicing and sets flags accordingly.
- Impact:** Represents the handoff to the next major business function of invoicing.

This documentation provides a comprehensive overview of the business logic implemented in the FO602R program, detailing the rules and conditions that govern order processing and selection.