LO101R.MBR

Path: NXCLOUD/rpgsrc/LO101R.MBR **Generated**: 2026-01-08 12:34:59 **Processing Time**: 15679ms

Business Logic for LO101R

This document outlines the business rules that govern the registration of purchase orders and associated details, based on an analysis of the RPG program LO101R. The primary focus is on the validation and processing of order header information, including checks against various reference data.

The core logic for processing purchase orders is contained within the main program logic of LO101R, which handles the retrieval and validation of order header details, updates to the order, and interaction with various subroutines for data validation.

Order Status and Header Rules

LO101R: LOHEL1, LOHELU, SIHEL1, RLTRL7, VVARL1, VLTYL1, VMVAL1, LODTPR, RP1PL1, SIHELA

1. Check for Existing Order Header

- **Logic:** The program checks if the order header exists in the LOHEL1 file. If found, it retrieves various order details.
- **File:** LOHEL1 (Order Header Register)
- **Field:** LOKOD
- **Condition:** The process will not proceed if the order header does not exist.

2. Validate Department Existence

- **Logic:** The program checks if the specified department exists in the department code register.
- **File:** KODE-REGISTER (Department Code Register)
- **Field:** C2AVDE
- **Condition:** The process will not proceed if the department does not exist.

3. Validate Buyer Existence

- **Logic:** The program checks if the specified buyer exists in the buyer code register.
- **File:** KODE-REGISTER (Buyer Code Register)
- **Field:** C2INNK
- **Condition:** The process will not proceed if the buyer does not exist.

4. Validate Warehouse Existence

- **Logic:** The program checks if the specified warehouse exists in the warehouse code register.
- **File:** KODE-REGISTER (Warehouse Code Register)
- **Field:** C2LAGE
- **Condition:** The process will not proceed if the warehouse does not exist.

5. Validate Payment Terms Existence

- **Logic:** The program checks if the specified payment terms exist in the payment terms register.
- **File:** KODE-REGISTER (Payment Terms Register)
- **Field:** C2BETB

•**Condition:** The process will not proceed if the payment terms do not exist.

Configuration and Authorization Rules

1. Check for Valid Invoice Issuer

•**Logic:** The program checks if the specified invoice issuer exists in the supplier register.

•**File:** RLEVL1 (Supplier Register)

•**Field:** C3FLEV

•**Condition:** The process will not proceed if the invoice issuer does not exist.

2. Check for Valid Project

•**Logic:** The program checks if the specified project exists in the project register.

•**File:** RP1PL1 (Project Register)

•**Field:** C3PROJ

•**Condition:** The process will not proceed if the project does not exist.

Financial and Transactional Rules

1. Check for Valid Invoice Number

•**Logic:** The program checks if the invoice number has been used previously for the supplier.

•**File:** SIHELA (Historical Invoice Register)

•**Field:** C3SFAK

•**Condition:** The process will not proceed if the invoice number has been used.

2. Check Invoice Dates

•**Logic:** The program checks if the invoice date is valid and not later than the due date.

•**File:** SIHELA (Historical Invoice Register)

•**Condition:** The process will not proceed if the invoice date is later than the due date.

Special Conditions (Program-Specific)

1. Check for Valid KID Number

•**Logic:** The program checks if the KID number is valid and performs a modulus check.

•**File:** SIHELA (Historical Invoice Register)

•**Field:** C3BKID

•**Condition:** The process will not proceed if the KID number is invalid. Note: The program prompts the user for confirmation if the KID number is not valid.

1. Update Order Header

•**Logic:** The program updates the order header with the validated details from the input fields.

•**File:** LOHELU (Order Header Update)

•**Condition:** The update occurs only if the order header exists.

Subprogram Calls Affecting Logic

1. CO402R (Check Warehouse)

•**Trigger:** Called during initialization to check if the warehouse differs from the user's default warehouse.

•**Logic:** Sets a flag if the warehouse is newly created and differs from the default.

•**Impact:** This affects how the department is retrieved from the warehouse.

2. RA507R (Department Query)

•**Trigger:** Called when querying department details.

•**Logic:** Retrieves department information based on the department code.

•**Impact:** Provides necessary validation for department existence.

3. RA509R (Buyer Query)

•**Trigger:** Called when querying buyer details.

•**Logic:** Retrieves buyer information based on the buyer code.

•**Impact:** Provides necessary validation for buyer existence.

4. RA510R (Warehouse Query)

•**Trigger:** Called when querying warehouse details.

•**Logic:** Retrieves warehouse information based on the warehouse code.

•**Impact:** Provides necessary validation for warehouse existence.

5. RA503R (Payment Terms Query)

•**Trigger:** Called when querying payment terms.

•**Logic:** Retrieves payment terms information based on the payment terms code.

•**Impact:** Provides necessary validation for payment terms existence.

6. RS740R, RS728R, RS770R, RS721R, RS722R (Various Queries)

•**Trigger:** Called for various checks and retrievals related to accounts, cost carriers, projects, and suppliers.

•**Logic:** Each subprogram retrieves relevant information based on the provided parameters.

•**Impact:** Ensures that all necessary validations are performed before proceeding with updates.

This documentation provides a comprehensive overview of the business logic implemented in the LO101R program, detailing the rules and validations that govern the processing of purchase orders.