AS100R.MBR

Path: NXCLOUD/rpgsrc/AS100R.MBR **Generated**: 2026-01-08 12:04:37 **Processing Time**: 13015ms

Business Logic for Number Series Calculation and Update

This document outlines the business rules that govern the calculation and updating of number series, based on an analysis of the RPG program AS100R. The primary focus is on how the program manages the retrieval and updating of number series based on various input parameters. The core logic for number series management is contained within the *entry subroutine in AS100R . The program processes requests to retrieve the next available number, return a previously used number, or validate a manually entered number against defined ranges.

Order Status and Header Rules

Number Series Management: anuml1, anumlur

1. Retrieve Standard Number

- **Logic:** If the input code (p_kode) is 0, the program retrieves the next available number in the automatic series.
- **File:** anuml1 (Number Register)
- **Field:** anuml1_key
- **Condition:** The program will not proceed if the number series does not exist (indicated by *in66 being *off).

2. Return Last Used Number

- **Logic:** If the input code (p_kode) is 1, the program returns the last used number to the number series.
- **File:** anumlur (Updated Number Register)
- **Field:** ansist
- **Condition:** The program will only update if the last used number matches the current number being returned.

Configuration and Authorization Rules

1. Manual Number Check

- **Logic:** If the input code (p_kode) is 2, the program checks if the manually entered number is within the defined series range.
- **Files:**
- anuml1 (Number Register)
- anumlur (Updated Number Register)
- **Fields:**
- p_numm (Manual Number)
- anntom (Upper Limit)
- **Condition:** The manual number must be greater than or equal to annfom and less than or equal to anntom. If outside this range, p_kode is set to 7.

2. Automatic Number Retrieval

- **Logic:** The program calculates the next number to be used in the automatic series.
- **File:** anumlur (Updated Number Register)
- **Field:** p_numm
- **Condition:** If the calculated number exceeds anntom, it wraps around to annfom if anwrap is set to 1, otherwise it sets p_kode to 8.

Financial and Transactional Rules

1. Update Number Register

- **Logic:** The program updates the number register with the new number if the input code is 0.
- **File:** anumlur (Updated Number Register)
- **Fields:**
- p_numm (New Number)
- ansist (Last Used Number)
- **Condition:** The update occurs only if the record is found (*in66 is *off).

2. Return Last Used Number Logic

- **Logic:** The program checks if the number being returned is the last used number and adjusts accordingly.
- **File:** anumlur (Updated Number Register)
- **Condition:** If the number being returned is not the last used number, it retains the last used number.

Special Conditions (Program-Specific)

1. Initialization Subroutine (AS100R)

- **Logic:** Initializes the program by setting up keys for the number register and loading parameters from the calling program.
- **File:** anuml1 (Number Register)
- **Field:** anuml1_key
- **Condition:** The initialization occurs upon entry into the program.

2. Manual Number Validation (AS100R)

- **Logic:** Validates whether a manually entered number is within the defined series limits.
- **File:** anuml1 (Number Register)
- **Fields:** p_numm (Manual Number), anntom (Upper Limit)
- **Condition:** If the manual number is outside the defined range, p_kode is set to 7.

Subprogram Calls Affecting Logic

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

1. mannnr (Manual Number Check)

- **Trigger:** Called when p_kode is 2.
- **Logic:** Checks if the manually entered number is within the defined range.
- **Impact:** This call determines if the manual number is valid and updates p_kode accordingly.

2. nyttnr (Next Number Calculation)

- **Trigger:** Called when p_kode is 0.
- **Logic:** Calculates the next available number in the automatic series.
- **Impact:** This is a critical step for number retrieval, affecting the output number returned to the calling program.

3. gammnr (Return Last Used Number)

- **Trigger:** Called when p_kode is 1.
- **Logic:** Determines the last used number and prepares it for return.
- **Impact:** This step ensures that the correct number is returned to the number series, maintaining integrity in number usage.