

AS100R.MBR

Path: NXCLOUD/rpgsrc/AS100R.MBR **Generated:** 2026-01-08 12:34:28 **Processing Time:** 12336ms

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 number series based on different operational codes.

The core logic for number series management is contained within the *entry subroutine in AS100R. The program processes requests to either retrieve the next available number in a series, return a previously used number, or validate a manually entered number against the defined series.

Order Status and Header Rules

Number Series Management: anuml1, anumlur

1. Retrieve Standard Number

- Logic:** The program retrieves the next available number in an automatic series if the operation code is 0.
- File:** anumlur (Number Register)
- Field:** ansist
- Condition:** The process will not select a record if the number series is exhausted (p_numm > anntom).

2. Return Last Used Number

- Logic:** The program allows the last used number to be returned to the series if the operation code is 1.
- File:** anumlur (Number Register)
- Field:** ansist
- Condition:** The last used number can only be returned if it matches the current last used number.

Configuration and Authorization Rules

1. Manual Number Validation

- Logic:** The program checks if a manually entered number is within the defined series range.
- Files:**
 - anuml1 (Number Series Definition)
- Fields:**
 - p_numm (Manual Number)
 - anntom (Upper Limit of Series)
- Condition:** The manual number will be rejected if it is less than annfom or greater than anntom.

2. Auto Number Retrieval

- Logic:** The program calculates the next available number in an automatic series.
- File:** anumlur (Number Register)

- Field:** p_numm
 - Condition:** The next number is calculated unless the series has wrapped around and all numbers are used (p_numm = *zero).
-

Financial and Transactional Rules

1. Update Number Register

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

2. Return Last Used Number Logic

- Logic:** The program adjusts the last used number based on specific conditions when returning a number.
 - File:** anumlur (Number Register)
 - Condition:** The last used number is only adjusted if it is the same as the current last used number.
-

Special Conditions (Program-Specific)

1. Initialization Subroutine (AS100R)

- Logic:** Initializes the program and sets up the key fields for the number register.
- File:** anuml1 (Number Series Definition)
- Field:** w_firm
- Condition:** The firm number is set from the input parameters.

2. Check for Number Series Existence (AS100R)

- Logic:** The program checks if a number series exists before proceeding with operations.
 - File:** anuml1 (Number Series Definition)
 - Fields:** anuml1_fell (Series Identifier), anuml1_type (Type of Series)
 - Condition:** The program sets p_kode to 9 if no series is found.
-

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 the operation code is 2 to validate a manually entered number.
- Logic:** Checks if the manually entered number is within the defined range.
- Impact:** This call acts as a **validation step** for manual entries.

2. nyttnr (Next Number Calculation)

- Trigger:** Called when the operation code is 0 to retrieve the next available number.
- Logic:** Calculates the next number based on the last used number and wraps around if necessary.

- Impact:** This is a **critical calculation step** for automatic number retrieval.

3. gammnr (Return Last Number)

- Trigger:** Called when the operation code is 1 to return the last used number.

- Logic:** Adjusts the last used number based on specific conditions.

- Impact:** This represents the **business logic for managing number returns**.