AA000R.MBR

Path: NXCLOUD/rpgsrc/AA000R.MBR **Generated**: 2026-01-08 12:22:41 **Processing Time**: 10474ms

–

Business Logic for User Registration Lookup

This document outlines the business rules that govern the user registration lookup process, based on an analysis of the RPG program AA000R. The primary focus is on how user records are accessed and validated within the system.

The core logic for user lookup is contained within the main processing section of the AA000R program. This program retrieves user records from the ausrl1 file and performs various checks to determine the validity of the user.

–

User Record and Validation Rules

User Lookup: ausrl1

1. User Record Retrieval

•**Logic:** The program attempts to retrieve a user record based on the provided user identifier. If the record is found, the program sets various data structure fields with the user's information.

•**File:** ausrl1 (User Registration File)

•**Field:** ausrl1_user

•**Condition:** The process will not proceed if the user record is not found (*in60 is set to *off).

2. User Validation Check

•**Logic:** If the user is not found, the program sets a flag (p_okok) to indicate an unsuccessful lookup and initializes the data structure (dsokok) to zero.

•**File:** ausrl1 (User Registration File)

•**Field:** p_okok

•**Condition:** The user lookup fails if the record is not found.

3. Special User Exemptions

•**Logic:** Certain users (e.g., 'ASPKASSE', 'NORGROS', 'ASPRMI') are exempt from the requirement to be defined in the user registration. If the user matches these values or if the first character of the user ID is 'Q', the lookup is considered unsuccessful.

•**File:** ausrl1 (User Registration File)

•**Field:** p_user

•**Condition:** The process will not select a record if p_user matches the specified exempt values or if w_alfa equals 'Q'.

–

Configuration and Initialization Rules

1. Program Initialization

•**Logic:** Upon program initialization, the p_okok variable is set to zero, indicating that no user has been successfully validated yet.

•**File:** N/A

•**Field:** p_okok

•**Condition:** This initialization occurs at the beginning of the program's execution.

2. User Parameter Passing

•**Logic:** The program accepts parameters for user identification and job context, which are used throughout the processing logic.

•**Files:**

•N/A

•**Fields:**

•p_user (User Identifier)

•p_jobb (Job Context)

•**Condition:** These parameters are mandatory for the program to execute properly.

–

Special Conditions (Program-Specific)

1. End of Program Logic

•**Logic:** The program sets the last record indicator (*inlr) to on, signaling the end of processing and allowing the program to terminate.

•**File:** N/A

•**Field:** *inlr

•**Condition:** This action is performed at the end of the program regardless of the outcome of the user lookup.

–

Subprogram Calls Affecting Logic

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

1. *inzsr (Initialization Subroutine)

•**Trigger:** This subroutine is called at the beginning of the program to initialize variables.

•**Logic:** It sets the p_okok variable to zero and prepares the program for user lookup.

•**Impact:** This initialization is crucial for ensuring that the program starts with a clean state.

2. *entry (Entry Point)

•**Trigger:** The program is invoked with parameters for user and job context.

•**Logic:** It processes the parameters passed to the program and sets up the necessary context for user validation.

•**Impact:** This entry point is essential for the program's operation, as it defines the parameters that will be used for user lookup.

3. chain (File Access Logic)

•**Trigger:** The chain operation is executed to access the user record in the ausrl1 file.

•**Logic:** It attempts to retrieve the user record based on the user identifier.

•**Impact:** This operation is a critical step in the user lookup process, determining whether the user exists in the system.

This documentation provides a comprehensive overview of the business logic implemented in the AA000R program, detailing the rules and conditions that govern user registration lookups.