FA120R.MBR

Path: NXCLOUD/rpgsrc/FA120R.MBR **Generated**: 2026-01-08 15:03:42 **Processing Time**: 12971ms

Business Logic for Job Register Maintenance

This document outlines the business rules that govern the Job Register Maintenance process, based on an analysis of the RPG program FA120R. The primary focus is on how job statuses are maintained and updated within the system.

The core logic for job status maintenance is contained within the *inzsr subroutine in FA120R. This subroutine initializes keys for reading and updating job records and user information, and it processes job status updates based on user interactions.

Job Status and Record Rules

Job Register Maintenance: fjobl1, fjoblu, fausrl1, ffa120d

1. Retrieve Job Information

•**Logic:** The program retrieves job information from the job register file fjobl1. If the record is found, it populates various job status fields; otherwise, it sets them to zero.

•**File:** fjobl1 (Job Register)

•**Field:** fajb01, fajb02, fajb03, fajb04, fajb05, fajb06, fajb07, fajb08, fajb09, fajb10

•**Condition:** The process will not select a record if *in90 is not set to *off.

2. Update Job Information

•**Logic:** The program updates the job register with new values based on user input. If the record does not exist, it writes a new record.

•**File:** fjoblu (Job Register Update)

•**Field:** fajb01, fajb02, fajb03, fajb04, fajb05, fajb06, fajb07, fajb08, fajb09, fajb10

•**Condition:** The update occurs if *in90 is set to *off.

User Interaction and Query Rules

1. User Query for Job Status

•**Logic:** The program allows users to query who started a job by checking specific fields based on user input. It retrieves user information from the user register.

•**Files:**

•fausrl1 (User Register)

•**Fields:**

•ausrl1_user (User ID)

•**Condition:** The query is triggered if the user presses the inquiry key (*inka).

2. Field-Specific User Lookup

•**Logic:** The program checks which field to search for user information based on the user's selection. It retrieves the corresponding user details and updates the display.

•**File:** fausrl1 (User Register)

•**Field:** ausrl1_user

•**Condition:** The lookup is performed based on the selected field (w_afld).

Special Conditions (Program-Specific)

1. Job Register Initialization (FA120R)

- **Logic:** Initializes keys for reading and updating job and user records. It sets the firm number in the key for job records.
- **File:** fjobl1, fjoblu, fausrl1 (Job and User Registers)
- **Field:** w_firm
- **Condition:** This initialization occurs at the start of the program.

2. User Lookup Subroutine (FA120R)

- **Logic:** This subroutine looks up user information from the user register based on the user ID provided. If found, it retrieves the user's name.
- **File:** fausrl1 (User Register)
- **Field:** w_bnav (User Name)
- **Condition:** The lookup is executed when the subroutine is called.

Subprogram Calls Affecting Logic

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

1. x_user (User Lookup Subroutine)

- **Trigger:** Called when user information is required based on job status fields.
- **Logic:** Looks up the user in the user register and retrieves the user name if the record exists.
- **Impact:** This call provides essential user context for job records, enhancing the clarity of job ownership.

2. *inzsr (Initialization Subroutine)

- **Trigger:** Automatically called at the beginning of the program.
- **Logic:** Initializes keys for job and user records, setting up the necessary context for subsequent operations.
- **Impact:** This is a foundational step that ensures all keys are correctly set before any processing occurs.

3. exfmt (Display Format Subroutine)

- **Trigger:** Called to display the job status and user information on the screen.
- **Logic:** Formats and presents the data to the user interface.
- **Impact:** This represents a critical point of interaction for users, allowing them to view and assess job statuses effectively.